Energy Efficient Robust Scheduling of Periodic Sensor Packets for Discrete Rate based Wireless Networked Control Systems

by

Bakhtiyar Farayev

A Thesis Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in

Electrical and Electronics Engineering

Koç University

September 4, 2015

Koç University
Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Bakhtiyar Farayev

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

Assist. Prof. Sinem Çöleri Ergen (Advisor)

Assoc. Prof. Öznur Özkasap

Prof. Özgür Erçetin

Date:

To my family

ABSTRACT

Wireless networked control systems (WNCS) require the design of a robust scheduling algorithm that meets the stringent timing and reliability requirements of control systems, despite the limited battery resources of the sensor nodes and the adverse properties of wireless communication for delay and packet errors. In this thesis, we propose a robust delay and energy constrained scheduling algorithm based on the exploitation of the periodic data generation nature of the sensor nodes at various pre-known frequencies in many WNCS applications such as smart grid and industrial automation. We first formulate the joint optimization of scheduling, power control and rate adaptation for practical discrete rate transmission model, in which only a finite set of transmission rates are supported, as a Mixed-Integer Non-linear Programming problem and prove its NP-hardness. We then propose a novel optimal polynomial-time power control and rate adaptation algorithm, and a novel polynomial-time heuristic scheduling algorithm based on determining the set of concurrently transmitting nodes by an intelligent iterative combining mechanism and distributing them over the schedule in the most uniform manner by a modified Karmarkar-Karp algorithm. The simulations demonstrate the superior performance of the proposed scheduling algorithm compared to previously proposed algorithms with lower runtime.

ÖZETÇE

Kablosuz Ağ Kontrol Sistemleri kablosuz algılayıcıların sınırlı enerji kaynağı ve kablosuz haberleşmenin gecikme ve paket hataları gibi olumsuz özelliklerine rağmen sıkı zamanlama ve güvenilirlik gerekliliklerini sağlayan etkin çizelgeleme tekniği gerektirmektedir. Bu tezde, akıllı örgü ve endüstriyel otomasyon gibi bir çok kablosuz ağ kontrol sistemlerinde algılayıcıların önceden bilinen frekanslarda periyodik paket üretme özelliğini kullanarak gecikme ve enerji gerekliliklerini sağlayan algoritma önermekteyiz. Biz ilk olarak çizelgeleme, güç kontrolü ve haberleşme hızı adaptasyonunu, gerçek sistemlerde olduğu gibi ayrık hız iletişim modelini kullanarak, birleşik bir biçimde Karışık Tümsayılı Lineer Olmayan Programlama problemi olarak formüle edip çözümünün NP-zorluğa sahip olduğunu gösteriyoruz. Daha sonra ise polinom zamanda çözülebilen güç kontrolü ve haberleşme hızı adaptasyonu algoritması, bunun yanı sıra yine polinomsal zamanda çözülebilen, eşzamanlı haberleşmeye uygun algılayıcıları akıllı bir döngüsel yöntemle birleştiren ve yapısı değiştirilmiş Karmarkar-Karp algorıtması ile çizelge üzerinde eşit dağılımlı bir şekilde dağıtmaya dayalı çizelgeleme problemi öneriyoruz. Yaptığımız benzetimle önerdiğimiz çizelgeleme algoritmasının önceki algoritmalara nazaran daha üstün performans sergilediğini ve daha düşük hesaplama süresine sahip olduğunu göstermektedir.

ACKNOWLEDGMENTS

Completing the masters degree is truly a long journey that would not be possible to end successfully without the help and support of many people. First and foremost, I am thankful my supervisor Assist. Prof. Sinem Çöleri Ergen for her endless support and guidance in every step within these 2 years.

I am also grateful to my committee members, Assoc. Prof. Öznur Özkasap and Prof. Özgür Erçetin for their comments, support and courtesy in serving as his defence committee.

Profound support from my family and my significant other Asya was also utmost important in my hard times. Lastly, I would love to express my gratitude to my friend and mentor Yalçın Şadi, with whom I collaborated on my research, Seyhan Uçar, who was always gave a hand in any problem, my office mates Saeed, Mohammad Ali, Sena, Yahya, Sanaz, Beakal, Tahseen, Utku, Merve, Anique, Elif for sharing the best times.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

The wireless communication between sensor nodes and controllers in WNCSs reduces the cost of their installation, maintenance and upgrade together with the complexity of the overall system compared to their wired equivalent [1]. WNCSs have been supported by several leading industrial organizations such as International Society of Automation (ISA) [2] and Highway Addressable Remote Transducer (HART) [3] with a vast number of applications in industrial automation [4], building automation [5], smart grid [6] and vehicle control systems [7].

The main challenge of WNCS is the design of a robust scheduling algorithm that satisfies the packet generation period, transmission delay and reliability requirements of the sensors required to maintain a certain control system performance, despite the adverse properties of wireless communication for non-zero packet error probability and non-zero delay at all times [8, 9]. The scheduling algorithm should exploit the periodic transmission nature of the sensor nodes at pre-known packet generation rates while providing robustness to packet losses and changes in network topology by distributing enough unused schedule periods over time allowing the inclusion of unexpected required additional packet transmissions as needed. Moreover, the schedule should satisfy the transmission energy requirement of the sensor nodes to achieve their desired lifetime, which are mostly battery operated or rely on energy harvesting techniques. Since the transmission delay and energy consumption of a sensor node are determined by the transmission power and rate of that sensor node and the concurrently transmitting nodes, the schedule should be optimized jointly with the transmission power and rate of the sensor nodes.

Scheduling algorithms designed for WNCS fall into one of two categories: robustness oriented and periodicity oriented. Scheduling algorithms designed for robustness aim to provide low deterministic end-to-end delay across a very large mesh network by adopting multi-channel Time Division Multiple Access (TDMA) together with multi-path multi-hop routing protocol [10, 11]. However, these algorithms schedule the packet transmissions one at a time without taking advantage of their predetermined periodic nature. Scheduling algorithms designed for periodicity on the other hand directly adopt the algorithms designed for the scheduling of periodic controller tasks running on a processor, such as Earliest Deadline First (EDF) and Least Laxity First (LLF), for the direct transmission of the periodic data packets of the sensor nodes to their corresponding controller [12, 13, 14]. However, these schedules allocate the transmissions as they arrive without leaving unused time periods in between failing to provide robustness to packet losses or topology changes in the network.

None of the schedules proposed for WNCS consider the assignment of the concurrent transmission of the sensor nodes to the time slots. Concurrent transmission reduces the overall schedule length by both decreasing the delay of the sensor nodes and increasing the length of available time intervals for retransmissions and unexpected transmission requests. Optimal concurrent transmission scheduling in a joint framework with power control and rate adaptation has been studied widely for delay constrained energy minimization in general purpose wireless networks [15, 16, 17]. However, none of these scheduling algorithms aim to satisfy the packet generation period and high reliability requirement of the sensors belonging to control systems by providing robustness to the packet losses in wireless communication.

Only recently, a novel framework for the design of a robust scheduling algorithm that exploits the periodicity of sensor node transmissions has been proposed for WNCSs [7, 18]. For the sake of completeness, we summarize the description and quantification of the robust schedule. The subframe and frame length are defined as the minimum and maximum packet generation period of all sensor nodes, and denoted by S and F, respectively. The *total active length of a subframe* m, denoted by a_m, is defined as the sum of the length of the time slots allocated to subframe m. The robustness of the schedule requires the distribution of data transmissions as uniformly as possible over the scheduling frame. The objective is therefore quantified as minimizing the maximum total active length of all the subframes in a frame.

We illustrate the robust schedule in comparison to EDF schedule with an example network consisting of 4 sensor nodes. Sensor node 1 has packet generation period of 1 ms and packet transmission time of $t_1 = 0.15$ ms. Sensor nodes 2, 3 and 4 have packet generation period of 2 ms and packet transmission times of $t_2 = 0.20$ ms, $t_3 = 0.25$ ms and $t_4 = 0.30$ ms, respectively. Here, the subframe and frame length are $S = 1$ ms and $F = 2$ ms, respectively. The frame contains 2 subframes. Fig. 1.1 (a) shows the robust schedule that minimizes the maximum total active length of the subframes. The maximum total active length is equal to $max(a_1, a_2) = max(t_1+t_2+t_3, t_1+t_4) = 0.60$ ms. Fig. 1.1 (b) on the other hand demonstrates the EDF schedule for the same network topology and sensor node requirements. The maximum total active length is equal to $max(a_1, a_2) = max(t_1+t_2+t_3+t_4, t_1) = 0.90$ ms. The robust schedule given in Fig. 1.1 (a) is more adaptive to packet losses, topology and sensor requirement changes than the EDF schedule given in Fig. 1.1 (b) as exemplified below:

- Suppose that the transmission of the data packet of sensor 1 in the first 1 ms failed. The robust schedule in Fig. 1.1 (a) includes enough space to allocate the retransmission of sensor 1 whereas the EDF schedule in Figure 1.1 (b) does not.

- Suppose that in addition to the periodic data packet generation of the scheduled sensor nodes, an additional packet of 0.2 ms time slot length is generated by an event-triggered sensor node at the beginning of the scheduling frame. Then the time slot of the event-triggered packet can be allocated with a delay of 0.60 ms in the schedule of Figure 1.1 (a) compared to 1.15 ms in the schedule of Figure 1.1 (b).

The robust schedules designed in [7] and [18] adopt the Ultra-Wideband trans-

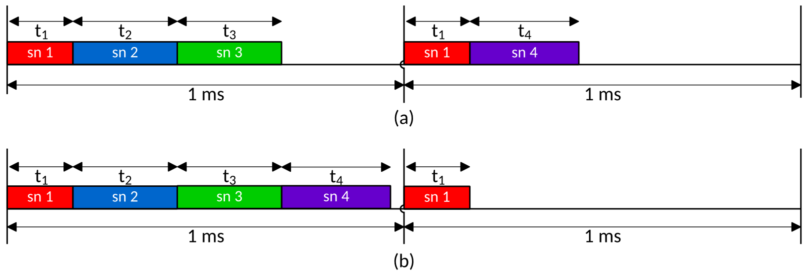

Figure 1.1: Schedules designed for the example network consisting of 4 sensor nodes, denoted by sn i for $i \in [1, 4]$. (a) Robust schedule that minimizes the maximum total active length of the subframes, (b) EDF schedule.

mission model, based on a linear relation between transmission rate and Signal-to-Interference-plus-Noise Ratio (SINR), due to very large bandwidth; and continuous rate transmission model, in which Shannon's channel capacity formulation for an Additive White Gaussian Noise (AWGN) wireless channel is used in the calculation of the maximum achievable rate as a function of SINR, respectively. In practical communication systems, however, only a finite set of discrete transmission rate values are supported, as suggested by the practical realization of multiple data rates in [19]. In this thesis, we study the joint optimization of scheduling, power control and rate adaptation with the objective of maximum robustness while meeting the periodic packet generation, transmission delay, reliability and energy consumption requirements of the sensor nodes adopting this discrete rate transmission model.

1.1 Contributions

The original contributions of this thesis are as follows:

- We study the joint optimization of power control, rate adaptation and scheduling of WNCSs based on the more realistic discrete rate transmission model for the first time in literature. We formulate the optimization as a Mixed-Integer Non-linear Programming (MINLP) problem and prove its NP-hardness. We then prove that the optimization of power control and rate adaptation for each node subset can be separately formulated, solved and then used in the scheduling algorithm in the optimal solution.

- We propose an optimal polynomial-time algorithm for a novel power control and rate adaptation problem with the objective of minimizing the time required for the concurrent transmission of a set of sensor nodes while satisfying their transmission delay, reliability and energy consumption requirements based on the more realistic discrete rate transmission model for the first time in literature. We provide a comprehensive proof of the optimality of this algorithm.

- We propose a novel heuristic polynomial-time scheduling algorithm based on first determining the set of concurrently transmitting nodes by an intelligent iterative combining mechanism using the proposed optimal power control and rate adaptation algorithm, and then distributing them over the schedule in the most uniform manner by a modified Karmarkar-Karp algorithm. Previously proposed algorithms mostly adopted the approach of first distributing node transmissions to the subframes with the goal of minimizing maximum total active length and then considering the concurrent transmission of the nodes allocated to the same subframe to decrease the total active length even further. Our approach of considering the concurrent transmission of the nodes before their distribution allows to allocate the node subsets better suited for concurrent transmission to the same subframe.

- We demonstrate that the proposed scheduling algorithm performs better than the previously proposed algorithms with lower runtime for different number of nodes and controllers in the network, different environments and node densities via extensive simulations.

1.2 Organization

The rest of the thesis is organized as follows. Chapter 2 describes the system model and assumptions used throughout this thesis. Chapter 3 presents the formulation of the joint optimization of power control, rate adaptation and scheduling of WNCSs as a MINLP problem, the proof of its NP-hardness and the reasoning for the solution strategy used in this thesis. Chapter 4 gives the optimal polynomial-time power control and rate adaptation algorithm and the proof of its optimality. Chapter 5 describes the proposed scheduling algorithm. Simulation results are presented in Chapter 6. The concluding remarks and the ideas for future research are given in Chapter 7.

Chapter 2

SYSTEM MODEL AND ASSUMPTIONS

The system model and assumptions are given as follows:

- The WNCS contains sensor nodes, controllers and actuators. The sensor nodes fall into one of two categories: time-triggered sensors and event-driven sensors. Time-triggered sensor nodes sample and send their data to one of the controllers periodically. The packet generation period and transmission delay required to achieve a certain control performance are denoted by T_l and d_l, respectively, for a time-triggered sensor node l. On the other hand, event-driven sensor nodes sample and send the data to the corresponding controller upon a triggering event in the unallocated parts of the schedule. The controllers can receive packets from one sensor node at a time. When a controller receives the sensor data, it performs the new control computation and sends the output to the actuator. We assume that the actuators always receive the controller packets successfully.

- We use the terms node and link interchangeably, where link l refers to the connection between sensor node l and its controller.

- We assume that sensor nodes transmit their data packets to their corresponding controllers in one hop. The extension of the proposed framework to include routing for the multi-hop transmission of the packets from the sensor nodes to the controllers is out of scope of this thesis and subject to future work.

- Central controller is selected from the existing controllers in WNCS and responsible for network synchronization, resource allocation and scheduling of the sensor nodes. Central controller is assumed to have complete network topology, however the method for topology learning is out of scope of this thesis. The central controller is also responsible of monitoring the received power and the packet error rate of the links in the network to make reliable decisions for resource allocation and scheduling of the sensor nodes.

- TDMA is used as a Medium Access Control (MAC) protocol due to its superior delay and energy performance for the networks with predetermined topology and data generation pattern [20]. The time is divided into subframes which are further divided into beacon and time slots. The central controller sends the beacon at the beginning of each subframe and includes the scheduling and resource allocation information of each scheduled node within the subframe. If there exist sensor nodes outside the transmission range of the central controller then the central controller uses the corresponding controllers to relay the beacon to these nodes. The guard time exists between consecutive time slots to compensate for synchronization errors.

- We assume that the packet generation periods are multiples of each other, e.g. $T = \{1, 2, 4, 8\}$ ms. The periodic generation of sensor data at various frequencies exists in many WNCS applications such as industrial automation [4], smart grid [6] and vehicle control [7].

- We assume that the length of the time slot allocated to a sensor node is fixed over all the subframes it has been allocated to and the time difference between the consecutive time slots assigned to each node is fixed and equal to its packet generation period. This assumption guarantees the fixed determinism of control systems, which is frequently preferred to bounded determinism [21].

- The nodes are operating in the active mode while transmitting and receiving the packets. They are set to sleep mode when the nodes are not scheduled to transmit or receive a packet. The nodes are in the transient mode when they are switching from active to sleep mode or vice versa. We only consider the energy consumption for the data packet transmission of each sensor node since the energy consumption in active mode is much larger than that in sleep and transient modes, and the energy consumption in the reception of the beacon packets is not subject to optimization. We define the maximum allowed per packet energy consumption to achieve a certain lifetime by e_l for sensor node l.

- The transmission power can take any value below P_{max}. This continuous power assumption is frequently used in previous scheduling algorithm designs since practical radios support a large number of discrete power levels, resulting in high approximation accuracy with lower complexity [22, 18, 17].

- We assume that the channel is constant over the scheduling frame, i.e. channel gain g_{lk} from the sensor node l to the controller receiving data from sensor node k does not change. Considering the channel variation of the links within the scheduling frame, i.e. fast fading wireless channel model, is out of scope of this thesis and subject to future work.

- We use the *discrete rate* transmission model in which a finite number of transmission rates $(r^1, r^2, ..., r^Q)$ corresponding to a finite number of SINR levels $(\gamma^1, \gamma^2, ..., \gamma^Q)$ are determined such that node l can transmit at rate r^q if the SINR achieved at the corresponding link, given by $\gamma_l = \frac{p_l g_{ll}}{N_0 + \sum_{l \neq k} p_l g_{lk}}$, is greater than or equal to γ^q, where p_l is the transmit power of node l, N_0 is the background noise power and Q is the number of discrete transmission rates. The transmission rates are assumed to be determined based on an SINR-rate mapping function $f(\gamma)$, i.e. $r^i = f(\gamma^i)$, $i \in [1, Q]$, such that $\frac{\partial^2 f(\gamma)}{\partial \gamma^2} \leq 0$. This property holds for Shannon's capacity formulation, which is commonly used in AWGN channels.

Chapter 3

OPTIMIZATION PROBLEM

3.1 Formulation of Optimization Problem

The joint optimization of power control, rate allocation and scheduling aims to provide maximum robustness while satisfying the packet generation period, transmission delay and energy requirement of the sensor nodes. This optimization problem is mathematically formulated as

minimize

$$t \tag{3.1a}$$

subject to

$$\sum_{n=1}^{N} t^{m,n} \leq t, \quad m \in [1, M] \tag{3.1b}$$

$$b_l^{m,n} t_l^{m,n} \leq t^{m,n}, \quad l \in [1, L], \; n \in [1, N], m \in [1, M] \tag{3.1c}$$

$$\sum_{m=m'}^{m'+s_l-1} \sum_{n=1}^{N} b_l^{m,n} = 1, \quad l \in [1, L], m' \in [1, M - s_l + 1] \tag{3.1d}$$

$$b_l^{m,n} = b_l^{m+s_l,n}, \quad l \in [1, L], n \in [1, N], m \in [1, M - s_l] \tag{3.1e}$$

$$\sum_{l \in L^c} b_l^{m,n} \leq 1, \quad c \in [1, C], n \in [1, N], m \in [1, M] \tag{3.1f}$$

$$t_l^{m,n} \leq d_l b_l^{m,n}, \quad l \in [1, L], n \in [1, N], m \in [1, M] \tag{3.1g}$$

$$t_l^{m,n} p_l^{m,n} \leq e_l b_l^{m,n}, \quad l \in [1, L], n \in [1, N], m \in [1, M] \tag{3.1h}$$

$$p_l^{m,n} \leq p_{max} b_l^{m,n}, \quad l \in [1, L], n \in [1, N], m \in [1, M] \tag{3.1i}$$

$$t_l^{m,n} x_l^{m,n} = F_l b_l^{m,n}, \quad l \in [1, L], n \in [1, N], m \in [1, M] \tag{3.1j}$$

$$p_l^{m,n} g_{ll} b_l^{m,n} - k_{lq} \gamma^q (N_0 + \sum_{u \neq l} p_u g_{ul} b_u^{m,n}) \geq 0, \quad l \in [1, L], n \in [1, N], m \in [1, M], q \in [1, Q] \tag{3.1k}$$

$$\sum_{q=1}^{Q} k_{lq}^{m,n} = 1, \quad l \in [1, L], n \in [1, N], m \in [1, M] \tag{3.1l}$$

$$x_l^{m,n} = \sum_{q=1}^{Q} k_{lq}^{m,n} r^q, \quad l \in [1, L], n \in [1, N], m \in [1, M] \tag{3.1m}$$

variables

$$t \geq 0,\, t^{m,n} \geq 0,\, t_l^{m,n} \geq 0,\, b_l^{m,n} \in \{0, 1\},\, p_l^{m,n} \geq 0,\, x_l^{m,n} \geq 0,\, k_{lq}^{m,n} \in \{0, 1\}$$
$$l \in [1, L], n \in [1, N], m \in [1, M], q \in [1, Q] \tag{3.1n}$$

where L is the number of sensor nodes; C is the number of controllers; N is the number of time slots in a subframe; M is the number of subframes in the scheduling frame given by the ratio of the frame length F to the subframe length S, i.e. $M = F/S$; L^c is the set of nodes transmitting to controller c; F_l is the length of the packet of sensor node l; and s_l is the ratio of the packet generation period of node l, i.e. T_l, to the subframe length S. The variables of the optimization problem are t, the maximum total active length of the subframes; $t^{m,n}$, the length of time slot n in subframe m; $t_l^{m,n}$, the transmission time of sensor node l in time slot n of subframe m; $b_l^{m,n}$, the indicator of the assignment of sensor node l to time slot n of subframe m; $p_l^{m,n}$, the transmit power of sensor node l in time slot n of subframe m; $x_l^{m,n}$, the transmission rate of sensor node l in time slot n of subframe m; and $k_{lq}^{m,n}$, the indicator of the assignment of sensor node l to the q-th rate level, i.e. r^q, in time slot n of subframe m.

Eqs. (3.1a) and (3.1b) are used to transform the objective of the optimization problem from a non-linear form of minimizing maximum total active of the subframes, i.e. $\max_{m\in[1,M]}\sum_{n=1}^{N} t^{m,n}$, to a linear form. Eq. (3.1c) states that the length of time slot n in subframe m is the maximum of the transmission times of the sensor nodes assigned to time slot n of subframe m. Eqs. (3.1d) and (3.1e) represent the periodic data generation requirement of the sensor nodes considering fixed determinism as stated in Section 2. Eq. (3.1f) states that the sensor nodes transmitting to the same controller cannot be allocated concurrently in a time slot. Eqs. (3.1g), (3.1h) and (3.1i) represent the transmission delay, transmission energy and maximum transmit power requirements of the sensor nodes, respectively. Eq. (3.1j) states that the amount of data transmitted by a sensor node must exceed its packet length. Eq. (3.1k) represents the relation between the transmission rate of sensor node l assigned to time slot n of subframe m and the transmission power of the sensor nodes allocated to the same time slot guaranteeing that the SINR is above the threshold corresponding to the assigned transmission rate of node l. Eqs. (3.1l) and (3.1m) guarantee that each link is assigned to one of the available discrete rate values.

The joint power control, rate adaptation and scheduling problem is a Mixed Integer Nonlinear Programming (MINLP) problem, for which there is no known polynomial-time algorithm.

3.2 NP-Hardness of Optimization Problem

Theorem 1. *The joint power control, rate adaptation and scheduling problem formulated in Section 3.1 is NP-Hard.*

Proof. Consider the instance of the original problem in which there exists only one controller in the WNCS. Then, only one node can be allocated to a time slot since the controller does not have the multi-reception capability. Therefore, the time slot length of each node can be calculated independently of the other nodes. Given the time slot length of all the sensor nodes, the problem reduces to the scheduling problem where the time slots corresponding to the sensor nodes are allocated to the subframes so that the maximum total active length is minimized while meeting the periodic packet generation requirement of the sensor nodes. This reduced form of the scheduling problem is the same as the problem introduced in Theorem 1 of [7], which proves the NP-Hardness of the scheduling problem for the one ECU case in IVWSNs by reducing the problem instance to the NP-hard Minimum Makespan Scheduling Problem (MSP) on identical machines. □

3.3 Solution Strategy

The solution strategy is based on the following Lemma.

Lemma 1. *In the joint optimization of power control, rate adaptation and scheduling, as formulated in Section 3.1, the optimization of the power control and rate adaptation of the node subsets with the goal of minimizing their concurrent transmission time, to minimize the total active length of the subframes, is independent of each other.*

Proof. Suppose that the set of nodes that are assigned to time slot n of subframe m for concurrent transmission are given so the values of the variables $b_l^{m,n}, l \in [1, L]$ satisfying Eqs. (3.1d) - (3.1f) are predetermined. Then the transmission power and transmission rate of the nodes in this set are determined by using Eqs. (3.1g)-(3.1m) with the goal of minimizing the length of the corresponding time slot given by Eq. (3.1c), independent of the other time slots. □

In our proposed solution, the optimization of power control and rate adaptation for each node subset is separately formulated, solved and then used in the scheduling algorithm as follows:

- We propose an optimal polynomial-time power control and rate adaptation algorithm with the objective of minimizing the time slot length required for the concurrent transmission of a set of sensor nodes while satisfying their transmission delay and energy requirements.

- Since solving an NP-hard problem optimally requires exponential runtime in the network size and is intractable even for moderate network sizes, in order to find fast and efficient solutions, we provide a novel heuristic polynomial-time scheduling algorithm based on determining the set of concurrently transmitting nodes and distributing them over the schedule in the most uniform manner by a modified Karmarkar-Karp algorithm. The algorithm for determining node subsets for concurrent transmission is based on iteratively combining the nodes that provide maximum gain in the total transmission time while limiting the size of the resulting node subsets considering the requirement for their uniform distribution in the following step. This approach has the advantage of determining better link subsets to be allocated to the same subframe in contrast to the previous algorithms that mostly adopted the approach of first distributing node transmissions to the subframes with the goal of minimizing the maximum total active length and then considering the concurrent transmission of the nodes allocated to the same subframe to decrease the total active length even further [7, 18].

Chapter 4

POWER CONTROL AND RATE ADAPTATION PROBLEM

The power control and rate adaptation problem aims to minimize the time required for the concurrent transmission of the sensor nodes, by assigning their transmission rates to one of the available discrete rate values, while satisfying their transmission delay, energy consumption and maximum allowed power constraints. The transmission rate, delay, energy and maximum power requirements are met by satisfying Eqs. (3.1g)-(3.1m) for a single time slot. Next, we propose a polynomial-time algorithm for the solution of this power control and rate adaptation problem and give a detailed proof of the optimality of the proposed algorithm.

4.1 Longest Transmission Time Rate Increase (LTTRI) Algorithm

We propose Longest Transmission Time Rate Increase (LTTRI) algorithm to determine the optimal power and rate allocation for the concurrent transmission of the nodes in the set S in polynomial time. The algorithm is based on the initialization of the transmission rate vector to the vector of the lowest possible rates satisfying the delay constraints of the nodes and then increasing the rate values in the transmission rate vector intelligently by checking the feasibility of the lowest possible number of rate allocation vectors for satisfying energy, maximum power and SINR constraints.

Let us enumerate the links in the set S from 1 to $|S|$. Let $\mathbf{r}$ be a vector of the transmission rates of the links in the set S. Let r_i denote the transmission rate of the i-th link, i.e. the i-th element of $\mathbf{r}$ vector; and γ_i refer to the corresponding SINR threshold level. Determining the feasibility of a rate vector $\mathbf{r}$ requires checking the existence of a power vector corresponding to $\mathbf{r}$; and if so, the satisfaction of the delay, energy and maximum power constraints by these rate and power vectors. The existence of a power vector corresponding to $\mathbf{r}$ is evaluated by determining the existence of a transmit power vector that satisfies the constraint

$$\frac{p_i g_{ii}}{N_0 + \sum_{j \neq i} p_j g_{ji}} > \gamma_i \tag{4.1}$$

for every node i. Perron-Frobenius conditions determine whether there exists a power vector satisfying these SINR constraints, and if so, the component-wise minimum power vector $\mathbf{p}^{min}$ [23]. $\mathbf{p}^{min}$ is then tested for delay, energy and maximum power constraints, given by Eqs. (3.1g), (3.1h) and (3.1i), respectively, for a single time slot.

Let t denote the time duration required for the concurrent transmission of the nodes in the set S. The optimal values of $\mathbf{r}$ and t are denoted by $\mathbf{r}^{opt}$ and t^{opt}, respectively. The i-th element of the vector $\mathbf{r}^{opt}$ corresponding to the optimal rate of node i denoted by $\mathbf{r}_i^{opt}$ is initialized to the minimum rate level that satisfies the delay constraint of node i, denoted by r_i^{th}, for all $i \in [1, |S|]$, whereas t^{opt} is initialized to infinity (Lines $1-2$). If this initial $\mathbf{r}^{opt}$ is not feasible, then the algorithm returns $t^{opt} = \infty$, meaning that there does not exist a feasible solution (Lines 3 and 15). Otherwise, the algorithm computes the duration of the time slot corresponding to the rate vector $\mathbf{r}^{opt}$ by determining the maximum transmission time of the nodes in the set S (Lines $4-5$). Since the node with the maximum transmission time denoted by j determines the time slot length, the length of the time slot can only decrease by increasing its transmission rate. Therefore, the algorithm continues by increasing the rate of the j-th node by one level if it is less than the maximum possible rate r^Q (Lines $6-7$) and then checking the feasibility of the resulting $\mathbf{r}^{opt}$ vector (Line 3). The algorithm stops if either the j-th node has rate r^Q, meaning that it is not possible to decrease the time slot length any further (Lines $8-9$) or the resulting $\mathbf{r}^{opt}$

vector is not feasible.

The complexity of LTTRI algorithm is $O(Q|S|^4)$ since the algorithm checks the feasibility of at most $Q|S|$ transmission rate vectors and the complexity of the feasibility check is dominated by that of testing Perron-Frobenius conditions given by $O(|S|^3)$ [23].

Algorithm 1 Longest Transmission Time Rate Increase (LTTRI) algorithm

Input: Node set S considered for concurrent transmission
Output: Optimal rate values for the nodes in the set S if their concurrent transmission is feasible

$t^{opt} = \infty$;
$\mathbf{r}^{opt} = (r_1^{th}, r_2^{th}, ..., r_{|S|}^{th})$,
  where $r_i^{th} = min_{q \in [1,Q]}\{r^q | \frac{F_i}{r^q} \leq d_i\}$;
while $\mathbf{r}^{opt}$ is feasible **do**
  $j = argmax_{i \in [1,|S|]} \frac{F_i}{r_i^{opt}}$;
  $t^{opt} = \frac{F_j}{r_j^{opt}}$;
  if $r_j^{opt} < r^Q$ **then**
    increase r_j^{opt} by one level;
  else
    break;
  end if
end while
if $t^{opt} < \infty$ and $\mathbf{r}^{opt}$ is not feasible **then**
  decrease r_j^{opt} by one level;
end if
return $\mathbf{r}^{opt}$, t^{opt}

4.2 *Optimality of LTTRI Algorithm*

We now prove the optimality of the proposed algorithm following the statement of the following three lemmas.

Let $\mathbf{r} = (r_1, r_2, ..., r_{|S|})$ be a vector of the transmission rates of the links in the set S and t be the length of the time slot corresponding to the rate vector $\mathbf{r}$, such that $t = max_{i \in [1,|S|]} \frac{F_i}{r_i}$, if $\mathbf{r}$ is feasible. Let us call the transmission rate vector $\mathbf{r}^d =$

$(r_1^d, r_2^d, ..., r_{|S|}^d)$, such that for at least one element k, $r_k^d < r_k$, while $r_i^d \leq r_i$ for all $i \in [1, |S|]$, the descendant of $\mathbf{r}$.

Lemma 2. *Let both $\mathbf{r}$ and $\mathbf{r}^d$ satisfy the delay constraint. If $\mathbf{r}^d$ is infeasible, then $\mathbf{r}$ is infeasible.*

Proof. Let us denote the SINR threshold levels corresponding to the rates r_i and r_i^d by γ_i and γ_i^d, respectively. Since higher rate corresponds to higher SINR threshold value and $r_i^d \leq r_i$, $\gamma_i^d \leq \gamma_i$. If $\mathbf{r}^d$ is infeasible since there does not exist a transmit power vector satisfying the SINR constraints given in Eq. (4.1) with threshold values of γ_i^d for all $i \in [1, |S|]$, then there does not exist a transmit power vector satisfying the SINR constraints with threshold values of γ_i for all $i \in [1, |S|]$, so $\mathbf{r}$ is infeasible.

Assume that there exist a transmit power vector corresponding to both $\mathbf{r}$ and $\mathbf{r}^d$. Let us denote the component-wise minimum power vectors corresponding to the rate vectors $\mathbf{r}$ and $\mathbf{r}^d$ by $\mathbf{p} = (p_1, p_2, ..., p_{|S|})$ and $\mathbf{p}^d = (p_1^d, p_2^d, ..., p_{|S|}^d)$, respectively. We first show that $p_i^d \leq p_i$ for all $i \in [1, |S|]$. Any power vector satisfying the SINR requirements for the SINR threshold levels $\gamma_i, i \in [1, |S|]$ also satisfies these requirements for $\gamma_i^d, i \in [1, |S|]$. Suppose that $p_k^d > p_k$ for an arbitrary link k. Since $\mathbf{p}$ satisfies the SINR requirements corresponding to the threshold levels $\gamma_i^d, i \in [1, |S|]$, $\mathbf{p}^d$ cannot be component-wise minimum power vector. This is a contradiction.

If $\mathbf{r}^d$ is infeasible due to the maximum power constraint, then there exists at least one link k for which $p_k^d > p_{max}$. Since $p_k \geq p_k^d$, $p_k > p_{max}$. Hence, $\mathbf{r}$ does not satisfy maximum power constraint either.

If $\mathbf{r}^d$ is infeasible due to the energy constraint, then there exists at least one link k for which $t_k^d p_k^d > e_k$. Let us denote the ratio of r_k to r_k^d by α. Since SINR-rate mapping function is an increasing function with a non-negative second derivative as described in Section 2, the transmission power p_k will be greater than αp_k^d. Therefore, $t_k p_k > \frac{F_k}{\alpha r_k^d} \alpha p_k^d = t_k^d p_k^d > e_k$ so $\mathbf{r}$ is infeasible due to the energy constraint. $\square$

Lemma 3. *Let t^d and t be the transmission times corresponding to the transmission rate vectors $\boldsymbol{r}^d$ and $\boldsymbol{r}$, respectively. Then, $t^d \geq t$.*

Proof. We know that, $t = max_{i \in [1,|S|]} t_i = max_{i \in [1,|S|]} \frac{F_i}{r_i}$. Since $r_i^d \leq r_i$ for all $i \in [1, |S|]$, $t = max_{i \in [1,|S|]} \frac{F_i}{r_i} \leq max_{i \in [1,|S|]} \frac{F_i}{r_i^d} = t^d$. Hence, $t^d \geq t$. □

Lemma 4. *Suppose that node j has the maximum time slot length among concurrently transmitting $|S|$ nodes, i.e. $j = argmax_{i \in [1,|S|]} \frac{F_i}{r_i}$. Then, increasing the transmission rate of any node $k \neq j$, where $k, j \in [1, |S|]$, does not decrease the time slot length t.*

Proof. $t = max_{i \in [1,|S|]} \frac{F_i}{r_i} = \frac{F_j}{r_j} \geq \frac{F_k}{r_k}$ for all $k \neq j$, where $k, j \in [1, |S|]$. Suppose that, for an arbitrary link $k \neq j$, we increase r_k to $r_k{}'$. Then $\frac{F_k}{r_k{}'} \leq \frac{F_k}{r_k} \leq \frac{F_j}{r_j} = t$. Hence, increasing the transmission rate of any node $k \neq j$ does not decrease t. □

Theorem 2. *Algorithm given in Section 4.1 gives the optimal transmission rate vector.*

Proof. We prove the theorem by showing that the transmission time corresponding to all the feasible transmission rate vectors is greater than or equal to the transmission time t^{opt} corresponding to $\mathbf{r}^{opt}$ at the output of the algorithm. We first partition the set of all possible transmission rate vectors into three subsets G_1, G_2 and G_3 such that $G_1 = \{\mathbf{r}' = (r_1', r_2', ..., r_{|S|}') | r_j' \leq r_j^{opt}; r_i' > 0, i \neq j\}$, $G_2 = \{\mathbf{r}'' = (r_1'', r_2''..., r_{|S|}'') | r_j'' \geq r_j^{opt,+}; r_k'' \geq r_k^{opt}, k \neq j\}$ and $G_3 = \{\mathbf{r}''' = (r_1''', r_2'''...r_{|S|}''') | r_j''' \geq r_j^{opt,+}; r_i''' < r_i^{opt}, i \in S_1, S_1 \neq \emptyset, r_i^{opt} > r^2; r_k''' \geq r_k^{opt}, k \in S_2, S_1 \cup S_2 \cup \{j\} = S, S_1 \cap S_2 = \emptyset, j \notin S_1, j \notin S_2\}$, where $j = argmax_{i \in [1,|S|]} \frac{F_i}{r_i^{opt}}$ and $r_j^{opt,+}$ is the rate increased by one level from r_j^{opt}. We now analyze the feasibility and the transmission time corresponding to all $\mathbf{r}' \in G_1$, $\mathbf{r}'' \in G_2$ and $\mathbf{r}''' \in G_3$.

(i) We prove that for all $\mathbf{r}' \in G_1$, the transmission time is greater than or equal to t^{opt}. We know that $j = argmax_{i \in [1,|S|]} \frac{F_i}{r_i^{opt}}$ and $t^{opt} = max_{i \in [1,|S|]} \frac{F_i}{r_i^{opt}} = \frac{F_j}{r_j^{opt}}$. Similarly, the transmission time t' corresponding to $\mathbf{r}'$ is given by $max_{i\in[1,|S|]}\frac{F_i}{r'_i} \geq \frac{F_j}{r'_j}$. Since $r'_j \leq r_j^{opt}$, $t' \geq \frac{F_j}{r_j^{opt}} = t^{opt}$.

(ii) We prove that all $\mathbf{r}'' \in G_2$ are infeasible. Since $j = argmax_{i\in[1,|S|]}\frac{F_i}{r_i^{opt}}$, the algorithm stops because either $r_j^{opt} = r^Q$ (Lines $8-10$ in Algorithm 1) or the rate vector $\mathbf{r}$ with $r_j = r_j^{opt,+}$ and $r_k = r_k^{opt}, k \in [1,|S|], k \neq j$ is infeasible (Line 3 in Algorithm 1). If $r_j^{opt} = r^Q$ then $G_2 = \emptyset$. Otherwise, since the rate vector $\mathbf{r}$ with $r_j = r_j^{opt,+}$ and $r_k = r_k^{opt}, k \in [1,|S|], k \neq j$ is a descendant of $\mathbf{r}''$, $\mathbf{r}''$ is also infeasible due to Lemma 2.

(iii) We prove that for all $\mathbf{r}''' \in G_3$, the transmission time is greater than or equal to t^{opt}. Since the rate vector $\mathbf{r}$ with $r_j = r_j^{opt,+}$ and $r_k = r_k^{opt}, k \in [1,|S|], k \neq j$ is infeasible, we need to decrease the rate of at least one node i such that $i \in [1,|S|], i \neq j, r_i^{opt} > r^2$ to guarantee feasibility. Let us assume that $\mathbf{r}''' = (r_1''', r_2'''...r_{|S|}''')$ for which $r_j''' \geq r_j^{opt,+}; r_i''' < r_i^{opt}, i \in S_1, S_1 \neq \emptyset, r_i^{opt} > r^2; r_k''' \geq r_k^{opt}, k \in S_2, S_1 \cup S_2 \cup \{j\} = S, S_1 \cap S_2 = \emptyset, j \notin S_1, j \notin S_2$ is feasible with the corresponding transmission time t'''. There exists only one descendant of $\mathbf{r}'''$, which is given by $\mathbf{r}^\beta = (r_1^\beta, r_2^\beta, ..., r_{|S|}^\beta)$ such that $r_i^\beta = r_i'''$ for at least one element i of the set S_1 and $r_k^\beta \leq r_k''', k \neq i, k \in [1,|S|]$, that Algorithm 1 evaluates with the i-th node having the maximum transmission time and continues by increasing r_i^β by one level. Since increasing the transmission rate of any node $k \neq i$ does not decrease the time slot length due to Lemma 4, $t^\beta = t'''$. Since $\mathbf{r}^\beta$ is a descendant of $\mathbf{r}^{opt}$, $t^\beta \geq t^{opt}$ due to Lemma 3. Therefore, $t''' \geq t^{opt}$. □

Chapter 5

SCHEDULING ALGORITHM

The scheduling algorithm aims to determine the concurrently transmitting node subsets and assign them to the subframes such that the maximum total active length of the subframes over a frame is minimized. The optimal scheduling has been formulated as a Mixed-Integer Linear Programming (MILP) problem with the number of variables exponential in the number of the links in [7]. The heuristic algorithm proposed in [7] is based on first sequentially assigning the time slot corresponding to each node to the subframe with the smallest total active length, and then iteratively improving the maximum total active length of the schedule by determining the node subset of maximum utility at each iteration, where the utility of a node subset is defined as the amount of decrease in the maximum total active length by the concurrent transmission of the nodes in the subset. [18] on the other hand proposes a framework for the design of a heuristic scheduling algorithm encompassing the previously proposed scheduling algorithm in [7]. This framework consists of a node assignment algorithm that assigns the nodes to the subframes in a certain order, and a concurrency allocation algorithm that determines the subsets of the nodes that can transmit concurrently from the node sets with the same packet generation period allocated to the same subframe by the node assignment algorithm. The main deficiency of these previously proposed algorithms is that the assignment of the nodes to the subframes does not consider the amount of decrease in their total transmission time by concurrent transmission: If the total transmission time of a subset of nodes decreases considerably by concurrent transmission, they should be assigned to the same subframe.

We propose a novel heuristic polynomial-time scheduling algorithm based on first determining the set of concurrently transmitting nodes, and then distributing them over the schedule in the most uniform manner by a modified Karmarkar-Karp algorithm. The two phases of the algorithm are called Adaptive Maximum Improvement based Concurrency (AMIC) Algorithm and Difference based Node Assignment (DNA) Algorithm, as depicted in Fig. 5.1. Since the proposed scheduling algorithm combines AMIC and DNA algorithms, we call it AMIC-DNA.

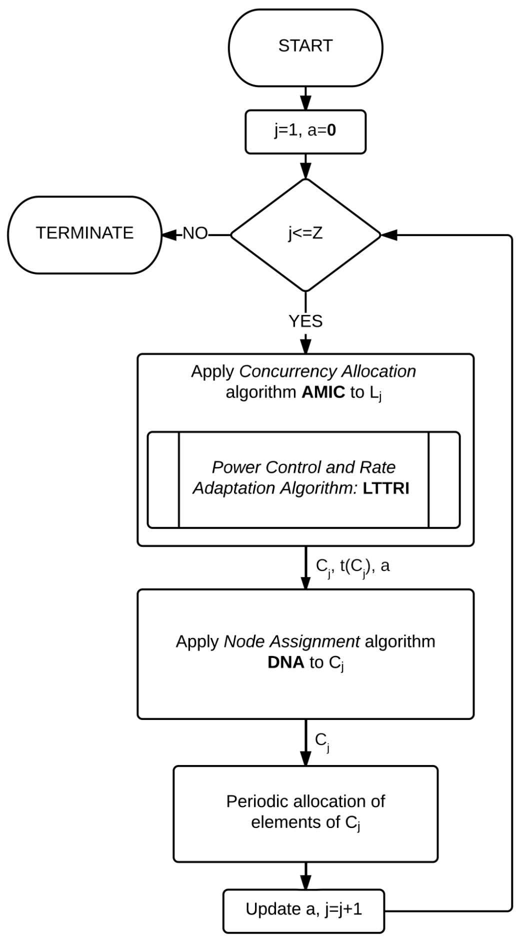

Figure 5.1: General framework of the AMIC-DNA algorithm.

Let L sensor nodes in the WNCS have Z distinct packet generation periods such that $T_1 < T_2 < ... < T_Z$. Let a be the vector of the total active length of M subframes in a frame. Let L_j denote the set of sensors with packet generation period T_j. AMIC and DNA algorithms are executed for each packet generation period T_j in the increasing order. We consider only the nodes with the same packet generation period for concurrent transmission to keep the time difference between consecutive time slot allocations fixed at their packet generation period for fixed determinism, as explained in detail in Section 2. AMIC algorithm partitions the set of sensor nodes with packet generation period T_j into concurrently transmitting node subsets. The subset of the nodes most suitable for concurrent transmission is determined by comparing the individual transmission time of the nodes with their concurrent transmission time. The feasibility and the transmission time of their concurrent transmission is determined by using LTTRI algorithm given in Section 4.1. The output of the AMIC algorithm includes the subsets of concurrently transmitting nodes, which is denoted by C_j, and their corresponding transmission time, called $t(C_j)$. The DNA algorithm then assigns the subsets in C_j to the subframes $k \in [1, s_j]$ in the most uniform manner so that the maximum total active length of the subframes is minimized. The resulting subsets in the first s_j subframes are then allocated periodically with period T_j over the frame to satisfy the periodicity requirement. a is initialized to zero vector at the beginning of the algorithm and updated at the end of the periodic allocation corresponding to each packet generation period.

5.1 Adaptive Maximum Improvement based Concurrency (AMIC) Algorithm

AMIC algorithm aims to partition the set of sensor nodes with packet generation period T_j, denoted by L_j, into concurrently transmitting node subsets by iteratively combining the node subsets with maximum gain quantified by the decrease in their total transmission time.

Since concurrency allocation is followed by their uniform distribution in the following DNA algorithm, an upper limit is imposed on the size of the subset of concurrently transmitting nodes as a function of the number of nodes in the set L_j and the number of subframes used for uniform distribution s_j. This limit is denoted by w and equal to $\lfloor \frac{|L_j|}{s_j} \rfloor$. If $w = 0$, then $|L_j| < s_j$, meaning that the number of nodes in L_j is less than the number of subframes s_j, so concurrency does not help distributing the nodes more uniformly and should be skipped. On the other hand, if $w \geq 1$, then this limit guarantees having at least s_j concurrently transmitting node subsets for uniform distribution. If such a limit is eliminated, at the extreme case, all the nodes in the set L_j may be combined for concurrent transmission in one subframe, which increases the maximum total active length considerably.

The metric used in the determination of concurrently transmitting node subsets is the decrease in the total transmission time of two node subsets when they transmit concurrently. Let $t(A)$ correspond to the transmission time of node subset A calculated by using LTTRI algorithm. Let L_j^1 and L_j^2 be two disjoint subsets of L_j. The gain by the concurrent transmission of L_j^1 and L_j^2 is then computed as

$$G(L_j^1, L_j^2) = t(L_j^1) + t(L_j^2) - t(L_j^1 \cup L_j^2) \tag{5.1}$$

AMIC algorithm, illustrated as Algorithm 2, is described as follows. The upper limit on the number of nodes in the concurrently transmitting node subsets w is initialized to $\lfloor \frac{|L_j|}{s_j} \rfloor$ (Line 1). The set of concurrently transmitting node subsets C_j and their transmission time $t(C_j)$ are initialized to the set of nodes L_j and their transmission time, respectively, meaning that initially every node is transmitting separately (Line 2). The algorithm then continues by combining the node subsets in C_j for concurrent transmission as long as there exist such two node subsets with the total

Algorithm 2 Adaptive Maximum Improvement based Concurrency Algorithm (AMIC)

Input: L_j : subset of nodes with packet generation period T_j.
Output: C_j : set containing subsets of concurrently transmitting nodes,
  $t(C_j)$: set of transmission times corresponding to subsets in C_j.

1: $w = \lfloor \frac{|L_j|}{s_j} \rfloor$;
2: $C_j = L_j$, $t(C_j) = t(L_j)$
3: **while** $\exists\, C_j^1 \in C_j$, $C_j^2 \in C_j$ s.t. $G(C_j^1, C_j^2) > 0$ and $|C_j^1| + |C_j^2| \leq w$ **do**
4: $\quad (C_j^{1*}, C_j^{2*}) = \text{argmax}_{C_j^1 \in C_j, C_j^2 \in C_j, |C_j^1| + |C_j^2| \leq w} G(C_j^1, C_j^2)$
5: $\quad C_j \leftarrow C_j - C_j^{1*} - C_j^{2*}$
6: $\quad t(C_j) \leftarrow t(C_j) - t(C_j^{1*}) - t(C_j^{2*})$
7: $\quad C_j \leftarrow C_j + (C_j^{1*} \cup C_j^{2*})$
8: $\quad t(C_j) \leftarrow t(C_j) + t(C_j^{1*} \cup C_j^{2*})$
9: **end while**

number of nodes in their union satisfying the upper limit constraint and their total transmission time decreasing by their concurrent transmission, i.e. their gain defined in Eq. (5.1) is greater zero (Line 3). In that case, the two node subsets with maximum gain are determined and subsequently replaced by their union in C_j, similarly their corresponding transmission times are replaced by that of their union in $t(C_j)$ (Lines $4-8$).

To illustrate the AMIC algorithm with an example, let us enumerate the nodes in L_j from 1 to 7 and set s_j to 2. The algorithm starts by setting w to 3, i.e. the maximum number of nodes allowed for concurrent transmission is 3, and C_j to $\{1, 2, 3, 4, 5, 6, 7\}$. Each iteration of the AMIC algorithm is shown in Table 5.1. In the first, second and third iteration of the algorithm, $(4, 7)$, $(3, 5)$ and $(2, 4, 7)$ are found to have the maximum gain, respectively. In the fourth iteration, there exist no subset pairs with positive gain, and thus the algorithm terminates.

The complexity of the AMIC algorithm is $O(Q|L_j|^7)$. The maximum number of iterations is $|L_j|$ since the number of elements decreases by 1 in each iteration. The algorithm checks all the node subset pairs in each iteration, with worst case complexity of $O(|L_j|^2)$. Calculating the gain corresponding to each such pair has

Table 5.1: Example concurrency allocation in AMIC algorithm.

Iteration	C_j	C_j^{1*}, C_j^{2*}
1	$\{1, 2, 3, 4, 5, 6, 7\}$	$4, 7$
2	$\{1, 2, 3, (4, 7), 5, 6\}$	$3, 5$
3	$\{1, 2, (3, 5), (4, 7), 6\}$	$2, (4, 7)$
4	$\{1, (2, 4, 7), (3, 5), 6\}$	$-$

$O(Q|L_j|^4)$ complexity, due to the complexity of the computation of the transmission time of their concurrent transmission by the LTTRI algorithm.

5.2 Difference based Node Assignment (DNA) Algorithm

DNA algorithm aims to distribute the concurrently transmitting node subsets C_j determined by the AMIC algorithm into the first s_j subframes such that the maximum total active length of the subframes is minimized.

This problem is similar to the NP-hard Minimum Makespan Scheduling Problem (MSP) on identical machines [7] and the NP-hard Multi-way Number Partitioning (MNP) problem [24, 25]. Given a set of n jobs with processing times $pt_i, i \in [1, n]$ and m identical machines, the MSP aims to find an assignment of the jobs to m identical machines such that the makespan, which is the time until all jobs have finished processing, is minimized. A 2-approximation algorithm for the MSP called List Scheduling Algorithm is a greedy algorithm based on assigning the jobs to the machines with the minimum current load [26]. On the other hand, given a multiset of positive integers, the MNP problem aims to assign each integer to one of k subsets, such that the largest sum of the integers assigned to any subset is minimized. Two main algorithms of polynomial-time complexity proposed for the MNP problem are greedy algorithm and Karmarkar-Karp (KK) algorithm [27], [25]. The greedy algorithm for this problem sorts the numbers in decreasing order and assign each number to the subset with the smaller sum so far. On the other hand, the KK algorithm is based on repeatedly assigning k largest numbers to different subsets, without determining which subset each goes into; combining them in the most efficient manner and replacing them by a difference measure, until no more such combining can be done. [25] and [28] shows that the KK algorithm has the best performance among all the polynomial-time alternatives for different numbers of elements in the original set via extensive simulations. Therefore, we use the adaptation of the KK algorithm in our proposed DNA algorithm.

The main difference between our problem, and the MSP and MNP problems, is the existence of the non-zero total active length of the subframes before the distribution of the concurrently transmitting node subsets by the DNA algorithm for packet generation period $T_i, i \in [2, Z]$. The vector of the total active length of the subframes a is initialized to zero before the distribution of the node transmissions for packet generation period T_1, and then updated by the DNA algorithm for all the packet generation periods $T_i, i \in [1, Z]$. We therefore adapt the KK algorithm to include an additional initial step to insert the vector a into the algorithm.

The DNA Algorithm, illustrated in Algorithm 3, is described as follows. Since the concurrently transmitting node subsets are distributed in the first s_j subframes, only the first s_j elements of the total active length vector a is stored in vector a' (Line 1). The set of concurrently transmitting node subsets, denoted by C_j, is then updated by including a'; the transmission time vector extended by including a' is renamed as $vt(C_j)$; and the difference measure of a' is calculated by the function $diff$ and included in the transmission time vector $t(C_j)$ (Lines $2-4$). The difference measure function $diff$ finds the sum of the differences between the smallest element and the remaining elements of a vector. For instance, $diff([1, 5, 4, 3]) = (5-1) + (4-1) + (3-1) = 9$. Throughout the algorithm, the vector $t(C_j)$ contains either the transmission times of the concurrently transmitting node subsets or the difference measure of the transmission time vectors whereas $vt(C_j)$ includes either the single

Algorithm 3 Difference Based Node Assignment Algorithm (DNA)

Input: C_j : set containing subsets of concurrently transmitting nodes,
$t(C_j)$: set of transmission times corresponding to subsets in C_j,
a : vector of total active length of the subframes.

Output: Allocation of the concurrently transmitting node subsets in C_j to the first s_j subframes.

$a' = [a_1, a_2, ..., a_{s_j}]$;
$C_j \leftarrow C_j + a'$;
$vt(C_j) \leftarrow t(C_j) + a'$;
$t(C_j) \leftarrow t(C_j) + diff(a')$;
while $|C_j| > 1$ **do**
  sort $t(C_j)$ in decreasing values, and correspondingly C_j and $vt(C_j)$;
  if $C_j(1 : s_j)$ contains combined element **then**
    $(C^*, tC^*) = \text{merge}(C_j(1 : s_j))$;
  else
    $(C^*, tC^*) = \text{combine}(C_j(1 : s_j))$;
  end if
  $C_j \leftarrow C_j - C_j(1 : s_j)$;
  $C_j \leftarrow C_j + C^*$
  $vt(C_j) \leftarrow vt(C_j) - vt(C_j)(1 : s_j)$;
  $vt(C_j) \leftarrow vt(C_j) + tC^*$
  $t(C_j) \leftarrow t(C_j) - t(C_j)(1 : s_j)$;
  $t(C_j) \leftarrow t(C_j) + diff(tC^*)$
end while
return C_j

values corresponding to the transmission times of the concurrently transmitting node subsets or the vectors of transmission times of length s_j.

The algorithm continues by sorting the elements of $t(C_j)$ in decreasing transmission time, and correspondingly those of C_j and $vt(C_j)$, and then merging or combining the first s_j elements at each iteration until there remains only one element in C_j (Lines $5-18$). Let us define the terms *combined element* and *singleton* before we proceed with the explanation of the functions *merge* and *combine*. A combined element is a vector of length s_j containing either numbers corresponding to a' or concurrently transmitting node subsets. A singleton is a concurrently transmitting node subset. If the first s_j elements of the vector C_j do not contain a combined element then they are combined by including them as elements of the vector C^* through the *combine* function (Lines $9-10$). For instance, if $s_j = 3$, and the first 3 elements of C_j are $(1,5),(2)$ and $(3,7)$ then $C^* = combine([(1,5),(2),(3,7)]) = [(1,5),(2),(3,7)]$. If the first s_j elements of the vector C_j contain at least one combined element then this combined element is merged with the remaining $s_j - 1$ elements, which may be either combined elements or singletons, through the *merge* function (Lines $7-8$). Merging two combined elements is assigning the largest entry of one combined element to the smallest entry of the other combined element. Merging a combined element with a singleton on the other hand is merely assigning the singleton to the smallest entry of the combined element. Once the combined element C^* is obtained by either *combine* or *merge* function, the first s_j entries of C_j are replaced by C^* (Lines $12-13$). Similarly, the vector of the total transmission time of C^*, denoted by tC^*, and the difference measure of tC^* replaces the first s_j entries of $vt(C_j)$ and $t(C_j)$, respectively (Lines $14-17$).

The complexity of the DNA algorithm is $O(|C_j|log(|C_j|))$. The algorithm initially sorts the entries of C_j with complexity $O(|C_j|log(|C_j|))$ and includes the difference measure of the combined element in the sorted list with complexity $O(log(|C_j|))$ for maximum $\frac{|C_j|}{s_j}$ iterations.

We can now illustrate the DNA algorithm with the following example. Let the input of the algorithm be $C_j = \{(1,2),(3),(4,7),(5),(6,9),(8),(10)\}$, $t(C_j) = \{1.1, 0.9, 1.5, 1.4, 1.65, 1.3, 1.05\}$, $a = [2.7, 2.6, 2.7, 2.6, 2.7, 2.6, 2.7, 2.6]$ ms and $s_j = 4$. This means that, we want to assign C_j to the first 4 subframes in a way that the resulting maximum total active transmission time of the first 4 subframes is minimized. The algorithm first constructs a' by taking the first s_j elements of a, i.e. $a' = [2.7, 2.6, 2.7, 2.6]$, and then includes a' in C_j and $vt(C_j)$, and its difference measure in $t(C_j)$. The resulting $vt(C_j)$ is given by $\{1.1, 0.9, 1.5, 1.4, 1.65, 1.3, 1.05, [2.7, 2.6, 2.7, 2.6]\}$. In the first iteration of the algorithm, first the entries of C_j, $vt(C_j)$ and $t(C_j)$ are sorted in decreasing values of $t(C_j)$. Since there is no combined element in the first 4 entries of C_j, the algorithm combines them generating $C^* = [(6,9),(4,7),(5),(8)]$ with difference measure calculated as $(1.65-1.3)+(1.5-1.3)+(1.4-1.3) = 0.65$, and updates C_j, $vt(C_j)$ and $t(C_j)$ accordingly. In the second iteration, following the sorting of the entries of C_j, $vt(C_j)$ and $t(C_j)$, the first 4 elements of C_j includes a combined element $[(6,9),(4,7),(5),(8)]$ with the corresponding $vt(C_j)$ value of $[1.65, 1.5, 1.4, 1.3]$. Hence, we merge $[(6,9),(4,7),(5),(8)]$ with the remaining 3 entries, $\{(1,2),(10),(3)\}$. The resulting combined element $C^* = \{(6,9);(4,7),(3);(5),(10);(8),(1,2)\}$ replaces these four entries in C_j. The transmission time values corresponding to C^* given by $[1.65, 1.5+0.9, 1.4+1.05, 1.3+1.1]$, and the difference measure $(2.45-1.65)+(2.4-1.65)+(2.4-1.65) = 2.3$ are included in $vt(C_j)$ and $t(C_j)$, respectively. In the third iteration, the remaining two combined elements are merged with the resulting $C^* = \{[2.6+(8),(1,2); 2.7+(5),(10); 2.6+(6,9); 2.7+(4,7),(3)]\}$. Since it contains only one element, the algorithm stops. The total active lengths of this final C_j vector is the single entry of $vt(C_j)$ given by $[5.1, 5.05, 4.35, 5]$. The difference measure of this partitioning is 2.1 ms, with maximum total active length equal to 5.1 ms.

Table 5.2: Example allocation of concurrently transmitting node subsets to the subframes by DNA algorithm.

step	C_j	$t(C_j)$
0,update/ 1,sorting	$\{(6,9),(4,7),(5),(8),(1,2),(10),(3),$ $[2.6,2.7,2.6,2.7]\}$	$\{1.65,1.5,1.4,1.3,1.1,1.05,0.9,0.2\}$
1,update/ 2,sorting	$\{(1,2),(10),(3),[(6,9),(4,7),(5),(8)],$ $[2.6,2.7,2.6,2.7]\}$	$\{1.1,1.05,0.9,0.65,0.2\}$
2,update/ 3,sorting	$\{[(6,9);(4,7),(3);(5),(10);(8),(1,2)],$ $[2.6,2.7,2.6,2.7]\}$	$\{2.3,0.2\}$
3,update	$\{[2.6+(8),(1,2);2.7+(5),(10);2.6+$ $(6,9);2.7+(4,7),(3)]\}$	$\{2.1\}$

The complexity of the algorithm for the node group belonging to each packet generation is dominated by the complexity of the AMIC algorithm, i.e. $O(Q|L_j|^7)$ for packet generation period T_j. Since $L_1 \cup L_2 \cup ... \cup L_Z$ forms the set of all the sensor nodes in the network and $O(Q|L_1|^7) + O(Q|L_2|^7) + ... + O(Q|L_Z|^7) \leq O(QL^7)$, the complexity of the AMIC-DNA algorithm is $O(QL^7)$.

Chapter 6

PERFORMANCE EVALUATION

The goal of the simulations is to evaluate the performance of the proposed power control and scheduling algorithm compared to the optimal solution, as well as the previously proposed scheduling algorithms. Since the scheduling algorithms proposed in [18] are shown to perform better than the Maximum Utility based Concurrency Allowance (MUCA) algorithm proposed in [7], we choose the best performing two algorithms using the scheduling framework proposed in [18]. The proposed scheduling framework assigns the nodes to the subframes through the node assignment algorithm and determines the subsets of nodes that concurrently transmit through the concurrency allocation algorithm. The first algorithm adopts Sorted Node Assignment (SNA) and Minimum Length Allocation (MLA) hence is denoted by SNA-MLA, whereas the second algorithm adopts SNA and Maximum Utility Allocation (MUA) therefore is called SNA-MUA. SNA is based on assigning each node to the subframe with the minimum total active length giving priority to the nodes with higher transmission times and considering the nodes connected to different controllers separately. MLA algorithm enumerates all feasible subsets of the nodes with the same packet generation period assigned to the same subframe, and then selects some of these feasible subsets such that every node is covered at least once and the total time allocated is minimized. MUA algorithm on the other hand iteratively constructs the concurrently transmitting node subsets from the nodes with the same packet generation period assigned to the same subframe by including the node that maximizes the utility, which is defined as the decrease in the transmission time of a set of sensor nodes when they transmit concurrently, in the subset. SNA-MLA with exponential complexity has been shown to perform better than SNA-MUA of polynomial-time complexity in [18]. To better understand the performance gain of the DNA algorithm compared to the greedy algorithm that assigns the node subsets with the largest transmit time to the shortest subframe in the scheduling, we also include the scheduling algorithm that combines AMIC and greedy algorithm, which is called AMIC-GREEDY. Furthermore, the optimal solution is obtained by enumerating all feasible solutions of the optimization problem and choosing the one with minimum maximum total active length for comparison and denoted by OPT.

Simulations are performed in MATLAB. The graphs show the average of 100 independent random network topologies, where the sensor nodes and the controllers are uniformly distributed within a square area. The packet generation period and the packet length of each node are randomly chosen from the sets $[1, 2, 4, 8]$ and $[50, 100]$ bits, respectively. 4 discrete rate levels corresponding to 4 SINR levels equal to $[-\infty, 10, 20, 30]$ dB are calculated by using Shannon's capacity formulation. The channel attenuation is calculated by using Rayleigh fading with scale parameter set to the mean power level determined by using the large scale statistics modeled as $PL(d) = PL(d_0) + 10\alpha \log(d/d_0) + Z$, where d is the distance between the node and the controller, $PL(d)$ is the path loss at distance d, $PL(d_0)$ is the path loss at reference distance d_0, Z is the Gaussian random variable with zero mean and standard deviation σ_z. All the simulation parameters are given in Table 6.1.

Table 6.1: Simulation Parameters

$PL(d_0)$	70 dB	p_{max}	250 mW
α	3.5	σ_z	4 dB
N_0	10^{-8} W/Hz	W	100 MHz

6.1 Maximum Total Active Length Performance

Fig. 6.1 shows the approximation ratio of the proposed AMIC-DNA algorithm compared to the AMIC-GREEDY, SNA-MLA and SNA-MUA algorithms for different number of nodes and controllers at 5 nodes/m^2 density. The approximation ratio is defined as the ratio of the maximum total active length of the scheduling algorithm to that of the optimal solution. The number next to the algorithm name in the legends of the figures denotes the number of controllers. AMIC-DNA algorithm outperforms both the previously proposed SNA-MLA and SNA-MUA algorithms, and AMIC-GREEDY algorithm for different number of nodes and controllers. The difference between AMIC-DNA and AMIC-GREEDY demonstrates the performance gain by the usage of the DNA algorithm whereas the remaining performance gain compared to the polynomial-time SNA-MUA algorithm quantifies the advantage of determining concurrently transmitting node subsets before their assignment to the subframes. The proposed AMIC-DNA algorithm even outperforms the exponential-time SNA-MLA algorithm. As the number of nodes and controllers increases, the approximation ratio slightly increases, mainly due to the increasing number of the combinations of the links for concurrent transmissions.

Fig. 6.2 shows the approximation ratio of the AMIC-DNA, AMIC-GREEDY, SNA-MLA and SNA-MUA algorithms for different path-loss exponents and different number of controllers in the network of 100 nodes. The figure again validates the superior performance of AMIC-DNA compared to AMIC-GREEDY, SNA-MLA and SNA-MUA algorithms for different path loss exponents. The approximation ratio only slightly increases with increasing path loss exponent, mainly due to the increase in the variance of link attenuations.

Fig. 6.3 shows the approximation ratio of the proposed and the previous algorithms for different network densities and different number of controllers in a net-

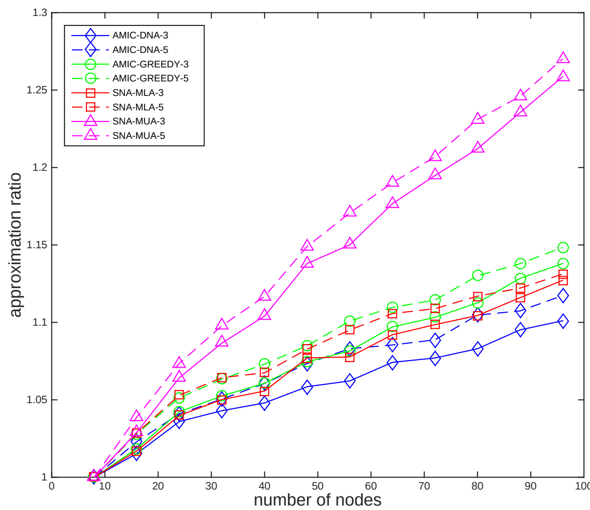

Figure 6.1: Approximation ratio of the scheduling algorithms for different number of nodes and controllers.

work of 100 nodes. This figure demonstrates that AMIC-DNA performs better than AMIC-GREEDY, SNA-MLA and SNA-MUA algorithms at different node densities. The difference between the performance of these algorithms decreases at low and high node densities. This is mainly due to the decreasing number of the combinations of the links for concurrent transmissions: At low node densities, the number of link combinations suitable for concurrent transmission is limited since most of the nodes are suitable for concurrent transmission while causing only slight interference being separated enough from each other. At high node densities, the number of node subsets that are separated enough from each other for concurrent transmission is limited due to the massive interference caused among each other.

6.2 Runtime Performance

Fig. 6.4 shows the average runtime of the scheduling algorithms for different number of nodes and controllers. The average runtime of the proposed AMIC-DNA algorithm

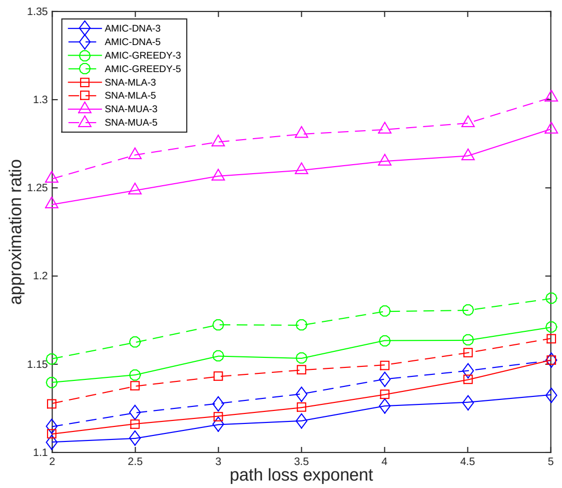

Figure 6.2: Approximation ratio of the scheduling algorithms for different path-loss exponents and different number of controllers in a network of 100 nodes.

increases almost linearly in the number of nodes, and only slightly increases as the number of controllers increases. As the number of controllers increases, the number of nodes that can concurrently transmit increases. This increases the complexity of the AMIC while decreasing that of DNA, assigning smaller number of subsets to the subframes. The average runtime of SNA-MUA algorithm is slightly smaller than that of AMIC-DNA for small number of controllers but starts to exceed that of AMIC as the number of controllers increases. As the number of controllers increases, the increase in the number of concurrently transmitting nodes increases the complexity of MUA while not affecting that of SNA. The average runtime of SNA-MLA increases exponentially in the number of the nodes, because of the exponential-time enumeration mechanism used in the MLA algorithm to determine the concurrent transmissions. The exponent of the runtime of both SNA-MLA and OPT increases as the number of controllers increases, with that of OPT larger than that of SNA-MLA.

When we analyze Figs. 6.1 and 6.4 together, we observe that AMIC-DNA provides

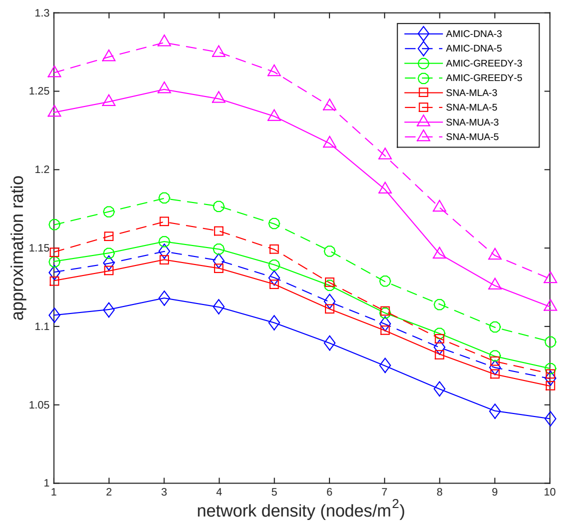

Figure 6.3: Approximation ratio of the scheduling algorithms for different network densities and different number of controllers in a network of 100 nodes.

superior maximum total active length performance at lower average runtime. This is mainly due to the consideration of concurrency allocation before the node assignment in the algorithm, and usage of efficient algorithms.

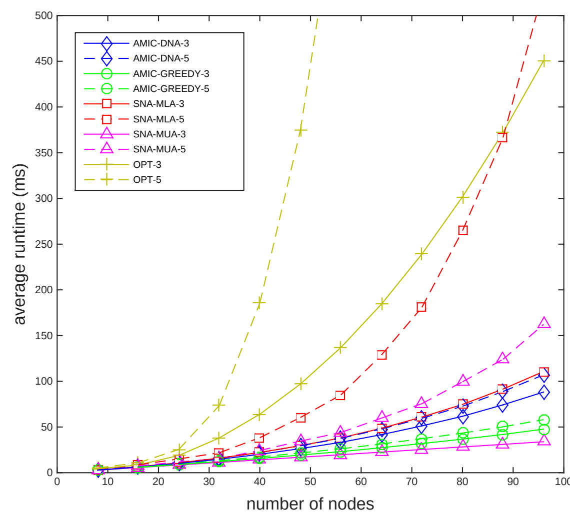

Figure 6.4: Average runtime of the scheduling algorithms for different number of nodes and controllers.

Chapter 7

CONCLUSION

In this thesis, we propose a novel solution method for the joint power control, rate adaptation and scheduling optimization with the goal of providing maximum robustness to the packet losses, topology and sensor requirement changes while satisfying the stringent timing and reliability requirements of control systems and energy requirements of the sensor nodes for practical WNCSs employing discrete rate transmission model. Upon formulation of the problem as a MINLP problem and proving its NP-hardness, we prove that the power control and rate adaptation problem can be separately solved and used in the scheduling algorithm in the optimal solution. We propose a polynomial-time algorithm for power control and rate adaptation, and prove its optimality. We then propose a novel heuristic scheduling algorithm where the concurrently transmitting node subsets are first determined with a limit on the maximum number of elements in each subset, and then distributed as uniformly as possible among the subframes by an adaptation of the Karmarkar-Karp algorithm, previously proposed for multi-way number partitioning problem. We demonstrate that the proposed algorithm performs better than the previously proposed algorithms with lower complexity for different network sizes, densities and environments. In the future, we are planning to extend this framework for multi-hop WNCS and next generation cellular networks.

BIBLIOGRAPHY

[1] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 138–162, January 2007.

[2] *ISA-100.11a-2009 Wireless systems for industrial automation: Process control and related applications*, ISA, 2009.

[3] *WirelessHART data sheet*, HART Communication Foundation, 2007, http://en.hartcomm.org/main_article/wirelesshart.html.

[4] A. Frotzsche, U. Wetzker, M. Bauer, M. Rentschler, M. Beyer, S. Elspass, and H. Klessig, "Requirements and current solutions of wireless communication in industrial automation," in *IEEE International Conference on Communications Workshops (ICC)*, June 2014, pp. 67–72.

[5] A. Aswani, N. Master, J. Taneja, D. Culler, and C. Tomlin, "Reducing transient and steady state electricity consumption in HVAC using learning-based model-predictive control," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 240–253, January 2012.

[6] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "A survey on smart grid potential applications and communication requirements," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 28–42, February 2013.

[7] Y. Sadi and S. Ergen, "Optimal power control, rate adaptation, and scheduling for UWB-based intravehicular wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 1, pp. 219–234, January 2013.

[8] Y. Wu, G. Buttazzo, E. Bini, and A. Cervin, "Parameter selection for real-time controllers in resource-constrained systems," *IEEE Transactions on Industrial Informatics*, vol. 6, no. 4, pp. 610–620, November 2010.

[9] W. P. M. H. Heemels, A. R. Teel, N. van de Wouw, and D. Nesic, "Networked control systems with communication constraints: Tradeoffs between transmission intervals, delays and performance," *IEEE Transactions on Automatic Control*, vol. 55, no. 8, pp. 1781–1796, August 2010.

[10] M. Baldi, R. Giacomelli, and G. Marchetto, "Time-driven access and forwarding for industrial wireless multihop networks," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 2, pp. 99–112, May 2009.

[11] B. Demirel, Z. Zou, P. Soldati, and M. Johansson, "Modular co-design of controllers and transmission schedules in wirelessHART," in *IEEE Conference on Decision and Control and European Control Conference*, Dec. 2011, pp. 5951–5958.

[12] P. Park, J. Araujo, and K. H. Johansson, "Wireless networked control system co-design," in *IEEE International Conference on Networking, Sensing and Control (ICNSC)*, April 2011, pp. 486–491.

[13] J. Bai, E. P. Eyisi, F. Qiu, Y. Xue, and X. D. Koutsoukos, "Optimal cross-layer design of sampling rate adaptation and network scheduling for wireless networked control systems," in *ACM/IEEE Third International Conference on Cyber-Physical Systems (ICCPS)*, April 2012, pp. 107–116.

[14] Y. Sadi, S. C. Ergen, and P. Park, "Minimum energy data transmission for wireless networked control systems," *IEEE Transactions on Wireless Communications*, vol. 13, no. 4, pp. 2163–2175, April 2014.

[15] A. Fu, E. Modiano, and J. N. Tsitsiklis, "Optimal transmission scheduling over a fading channel with energy and deadline constraints," *IEEE Transactions on Wireless Communications*, vol. 5, no. 3, pp. 630–641, March 2006.

[16] W. Chen, M. J. Neely, and U. Mitra, "Energy-efficient transmissions with individual packet delay constraints," *IEEE Transactions on Information Theory*, vol. 54, no. 5, pp. 2090–2109, May 2008.

[17] Y. Sadi and S. C. Ergen, "Minimum length scheduling with packet traffic demands in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 7, pp. 3738–3751, July 2014.

[18] Y. Sadi and S. Coleri Ergen, "Energy and delay constrained maximum adaptive schedule for wireless networked control systems," *IEEE Transactions on Wireless Communications*, accepted 2015.

[19] S. Cui, A. Goldsmith, and A. Bahai, "Energy-constrained modulation optimization," *IEEE Transactions on Wireless Communications*, vol. 4, no. 5, pp. 2349 – 2360, September 2005.

[20] S. Ergen and P. Varaiya, "PEDAMACS: power efficient and delay aware medium access protocol for sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 7, pp. 920–930, July 2006.

[21] J. R. Moyne and D. M. Tilbury, "The emergence of industrial control networks for

manufacturing control, diagnostics, and safety data," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 29–47, January 2007.

[22] S. Kompella, J. Wieselthier, A. Ephremides, H. Sherali, and G. Nguyen, "On optimal SINR-based scheduling in multihop wireless networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 6, pp. 1713 –1724, December 2010.

[23] R. A. Horn and C. R. Johnson., *Matrix Analysis.* Cambridge University Press, 1994.

[24] R. E. Korf, "A hybrid recursive multi-way number partitioning algorithm," in *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, vol. 22, no. 1, 2011, pp. 591–596.

[25] ——, "Multi-way number partitioning," in *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, 2009, pp. 538–543.

[26] R. Graham, "Bounds for certain multiprocessing anomalies," *The Bell System Technical Journal*, vol. 45, no. 9, pp. 1563–1581, November 1966.

[27] N. Karmarkar and R. M. Karp, "The difierencing method of set partitioning," Technical Report UCB/CSD 82/113, Computer Science Division, University of California, Berkeley, Tech. Rep., 1982.

[28] R. E. Korf, "From approximate to optimal solutions: A case study of number partitioning," in *IJCAI*, 1995, pp. 266–272.

VITA

BAKHTIYAR FARAYEV was born in Baku, Azerbaijan on 18th of November. He received his early education from Baku Turkish High School in Baku. He completed his B.Sc degree in Telecommunications Engineering in Istanbul Technical University (ITU), Istanbul, Turkey in 2013. In the same year he joined the M.Sc program in Electrical and Electronics Engineering at Koc University as a research and teaching assistant. During his study he worked on energy-efficient and robust communications in wireless sensor networks under the supervision of Assist. Prof. Sinem Çöleri Ergen. He has a seminal and a journal papers submitted to IEEE GLOBECOM 2015 Workshop and IEEE Transaction on Wireless Communication which both are under review at the moment.