Rollout Algorithm Based Duty Cycle Control with Joint Optimisation of Delay and Energy Efficiency for Beacon-enabled IEEE 802.15.4 Networks

Yun Li, Kok Keong Chai, Yue Chen
School of Electronic Engineering and Computer Science
Queen Mary University of London
Email: {yun.li, michael.chai, yue.chen}@qmul.ac.uk.

Jonathan Loo
Department of Computer and Communications Engineering
Middlesex University
Email: J.Loo@mdx.ac.uk

***Abstract*—Duty cycle control is applied in IEEE 802.15.4 medium access control (MAC) protocol to reduce energy consumption. A low duty cycle improves the energy efficiency but it reduces the available transmission time, thereby increases the end-to-end delay. Thus, it is a challenge issue to achieve a good trade-off between energy efficiency and delay. In this paper, we study a duty cycle control problem with the aim of minimising the joint-cost of energy consumption and end-to-end delay. By applying dynamic programming (DP), the optimal duty cycle control is derived. Furthermore, to ensure the feasibility of implementing the control on computation limited sensor devices, a low complexity rollout algorithm based duty cycle control (RADutyCon) is proposed. The joint-cost upper bound of the proposed RADutyCon is investigated. Simulation results show that RADutyCon can effectively reduces the joint-cost of energy consumption and end-to-end delay under various network traffic. In addition, RADutyCon achieves an exponential reduction of computation complexity compared with DP optimal control.**

I. Introduction

The emerging wireless sensor and actor networks (WSANs) are featured as integrating various applications and providing device-varied data delivery in terms of energy efficiency and end-to-end delay [1], [2]. IEEE 802.15.4 standard [3] utilises low duty cycle to conserve energy by putting devices into inactive mode. However, a lower duty cycle introduces higher end-to-end delay due to the reduced available transmission time. In addition, as application requirements various from device to device, the uniformed duty cycle control for all devices in current standard may not provide the best overall performance to meet the requirement for applications in WSANs.

The idea of achieving a trade-off between energy efficiency and end-to-end delay through adaptive duty cycle control of MAC protocols was explored by Dynamic Sensor MAC protocol (DSMAC) [4]. In DSMAC, duty cycle is adjusted based on the threshold of energy utilisation efficiency and average latency experienced by the sensor. However, the duty cycle adaptation of DSMAC can only be double or half of the initial setting. The delay reduction of U-MAC [5] is achieved by controlling the length of active periods based on a utilisation function, which is the ratio of the actual transmission and receptions performed by the device. However, the uniformed duty cycle control for all devices is not flexible when each device generates different amount of traffic with different quality of service (QoS) requirements. The duty cycle control algorithm called Traffic-adaptive Distance-based Duty Cycle Assignment (TDDCA) is proposed in [6], with the aim of meeting a target transmission rate while minimising the energy consumption. The duty cycle is increased when contention is reported. Otherwise, the duty cycle is decreased every time period down to a minimum. However, to enable the control, contention reports, piggyback flags and modifications of the packet header are needed. In [7], DutyCon is proposed to guarantee end-to-end delay by assigning a local delay requirement to each single hop along the communication flow. In this method, a feedback controller is designed to adapt the sleep interval to meet the single-hop delay requirement. However, this approach requires significant amount of signalling from the neighbour devices to compute the delay. To reduce the signalling among neighbour devices, a distributed duty cycle controller is proposed in [2] aiming at controlling the local queue length of the device to be the same as the predetermined threshold. The distributed duty cycle control is achieved by adjusting the sleep duration of each device based on its local queue length independently. However, this approach needs specific syntonisation scheme, and the evolution of the proposed control requires carefully setting of the initial duty cycle and control parameters.

While the aforementioned literatures laid a solid foundation in designing adaptive duty-cycled MAC protocols, less work has been done in terms of the duty cycle optimisation with joint consideration of energy consumption and end-to-end delay. To address the above joint consideration, in this paper we study an optimisation problem aiming at minimising energy consumption and end-to-end delay jointly. A joint-cost function, which follows a similar logic of the joint consideration of purchase cost and store cost in inventory control problem [8], is designed as the weighted sum of energy consumption and end-to-end delay. The weighting factors of the joint-cost function are adjustable according to different requirements on energy consumption and end-to-end delay of each device or specific application requirements.

The contributions of the paper are summarised as follows:

first, we formulate an optimisation problem to minimise the joint-cost, taking the network traffic and the device position into consideration. Then, the optimal duty cycle control is derived by applying dynamic programming (DP). Furthermore, a rollout algorithm based control (RADutyCon) is proposed to reduce the computation complexity of running DP on sensor devices. In addition, the joint-cost upper bound of the proposed RADutyCon is investigated.

The remainder of the paper is organised as follows. In section II, we give the system model and the background of IEEE 802.15.4 MAC protocol. Problem formulation is given in Section III. Section IV presents the derived optimal solution and the proposed RADutyCon. Simulation results and conclusion are given in section V and section VI, respectively.

II. System Model

Among the different multi-hop WSANs, the simple two-hop cluster-tree network model has been the focus of much ongoing research. This paper dedicated to the analysis of two-hop cluster-tree network while the multi-hop case can be viewed as the combination of several two-hop scenarios. We consider a three-level cluster-tree network as shown in Fig. 1. The coordinator n_0 is in level-1; the full-function devices (FFDs/actors) n_i ($1 \leq i \leq N$) are in level-2 and the reduced-function devices (RFDs/sensors) are in level-3. The level of the device is denoted as l_{n_i}, in particular, $l_{n_0} = 1$. FFDs can communicate with the coordinator and its child RFDs, whereas RFDs can only communicate with its parent FFD.

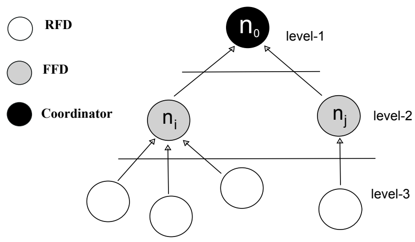

Fig. 1. Network Model.

A. IEEE 802.15.4 (2011)

We adopt IEEE 802.15.4 (2011) beacon-enabled mode where each FFD periodically broadcasts the beacon to its child devices. The duration between two consecutive beacons is called Beacon Interval (BI), while the duration of an active period is called Superframe Duration (SD). Specifically,

$$BI = aBaseSuperFrameDuration \times 2^{BO}, \quad (1)$$

$$SD = aBaseSuperFrameDuration \times 2^{SO}, \quad (2)$$

where Beacon Order (BO) and Superframe Order (SO) are two integers ranging from 0 to 14 ($0 \leq SO \leq BO \leq 14$), and $aBaseSuperFrameDuration = 15.36ms$ at 2.4 GHz with 250 kbps bandwidth. The duty cycle is defined as the ratio of the active portion over each time period, thus

$$Duty\ Cycle = SD/BI = 2^{SO-BO}. \quad (3)$$

In multi-hop transmission, each FFD divides its BI into two superframes, named incoming superframe and outgoing superframe, as shown in Fig. 2. The FFD n_i receives the beacon from the coordinator in the incoming superframe, and transmits its beacon in the outgoing superframe. As there are two SDs in each BI, according to (1) and (2), $SO \leq BO-1$ for all FFDs. Theoretically speaking, the duty cycle of different devices could be different, but in the current standard they are all equal [3].

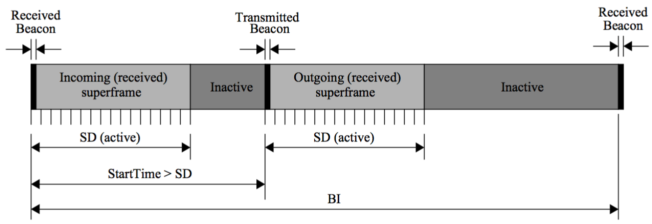

Fig. 2. Superframe structure of IEEE 802.15.4.

To simplify the problem, we aim at controlling the out going superframe duty cycle (refer as duty cycle in this paper) of the FFDs. The incoming superframe duty cycle is decided by the parent FFD of the device and enclosed in the received beacon. We set all devices to be activated at the beginning of each BI. The same BO is set to all devices in the network with the aim of simplifying the synchronisation. Thus, the duty cycle control of each FFD is achieved by setting the outgoing SO based on the number of packets generated by its child devices.

B. Queue and Traffic Models

We assume all generated packets are available at the beginning of each time period. All the packets are forwarded to the coordinator n_0 for uplink transmission and $q_{n_i}^{max}$ is the maximum queue length of the device n_i. The new arrived packets will be dropped if the queue length in the buffer reaches its maximum. Similar to [5], the queue length of time period $k+1$ of device n_i is given as

$$q_{n_i}^{k+1} = \min\left([q_{n_i}^k + r_{n_i}^k - f_{n_i}^k + g_{n_i}^k]^+, q_{n_i}^{max}\right), \quad (4)$$

where $0 \leq k \leq K-1$, $[\cdot]^+ = \max(0, \cdot)$, $g_{n_i}^k$ is the number of packets being generated by device n_i in time period k; $f_{n_i}^k$ is the number of packet transmited by device n_i in time period k; and $r_{n_i}^k$ is the number of packets received by device n_i in time period k. Note that $r_{n_i}^k$ equals to zero if device n_i has no child device. We assume the number of packets each device sends to its parent device follows Poisson distribution and each device generates a Poisson distributed integer number of packets in each time period (BI). Thus, $f_{n_i}^k$ and $g_{n_i}^k$ are independent random variables.

III. Problem Formulation

In this section, we formulate the duty cycle control problem as a dynamic programming inventory control problem to minimize the total expected joint-cost of energy consumption and end-to-end delay.

To minimise the total expected joint-cost of energy consumption and end-to-end delay, we define the transmitting energy consumption cost $E_t(f_{n_i}^k)$, receiving energy consumption cost $E_r(r_{n_i}^k)$, idle listening energy consumption cost $E_l(r_{n_i}^k)$, and end-to-end delay cost $D(r_{n_i}^k)$ of device n_i as

$$E_r(r_{n_i}^k) = c_r \times \frac{r_{n_i}^k}{q_{n_i}^{max} \times l_{n_i}} \tag{5}$$

$$E_t(f_{n_i}^k) = c_f \times \frac{f_{n_i}^k}{q_{n_i}^{max} \times l_{n_i}}, \tag{6}$$

$$E_l(r_{n_i}^k) = c_l \times \frac{[f_{n_i}^k - g_{n_i}^k - q_{n_i}^k - r_{n_i}^k]^+}{q_{n_i}^{max} \times l_{n_i}}, \tag{7}$$

$$D(r_{n_i}^k) = c_d \times \frac{[q_{n_i}^k + r_{n_i}^k + g_{n_i}^k - f_{n_i}^k]^+}{q_{n_i}^{max} \times l_{n_i}}, \tag{8}$$

where c_f, c_r, c_l and c_d are the coefficients of transmitting, receiving, idle listening and delay of the device, respectively. Note that $c_r < c_l$, as if c_r were greater than c_l, it would never be optimal to receive new packets in the last period and possibly in earlier periods.

We further introduce α and β to assign the weightings of energy efficiency and end-to-end delay requirements of different applications. The expected weighted-sum joint-cost function for device n_i at time period k is

$$J(r_{n_i}^k) = \mathbb{E}\Bigg\{\alpha\Bigg(E_f(f_{n_i}^k) + E_r(r_{n_i}^k) + E_l(r_{n_i}^k)\Bigg) + \beta D(r_{n_i}^k)\Bigg\}. \tag{9}$$

where $3\alpha + \beta = 1$, as there are three terms using the weighing factor α.

We adopt IEEE 802.15.4 (2011) standard, which applies slotted carrier-sense multiple access with slotted collision avoidance (CSMA/CA) for packet transmission. Before the packet transmission, we assume devices need to perform two clear channel accesses (CCAs). Within each superframe duration, the beacon transmission duration is D_{bcn}. Thus, the total packet transmission duration $PD = SD - D_{bcn}$. If acknowledgement (ACK) is required for each packet, the successful packet transmission period $P_s = \lceil P_{CCA} + P_L + \delta + P_{ACK} \rceil$, where P_{CCA} is the transmission time for two CCAs, P_L is the transmission time for each packet, δ and P_{ACK} are waiting and transmission time of the ACK packet, respectively. Hence, the number of packets that can be received by device n_i at k^{th} time period $r_{n_i}^k = PD/P_s$.

Because of the collision, the transmission throughput is limited according to the number of contending devices. We adopt the throughput limitation coefficient b of [9]. Based on (2), the relationship between SO and the amount of packets the device could receive in time period k is given as follow

$$SO_{n_i}(k) = \left\lceil \log_2\left(\frac{r_{n_i}^k \times P_s}{b} + D_{bcn}\right) \right\rceil. \tag{10}$$

Our objective is to find the control of the optimal duty cycles $\pi_{n_i}^*$ for each device n_i over K time periods, which minimise the overall expected joint-cost. Hence, the joint optimisation problem is:

$$\begin{aligned} \mathcal{P}_{n_i}: \quad & \min_{\pi_{n_i} \in \mathfrak{D}} \quad \mathbb{E}\left\{\sum_{k=0}^{K-1} J(r_{n_i}^k)\right\} \\ & \text{s.t.} \quad q_{n_i}^K = 0, \\ & \qquad\; r_{n_i}^k \le r_{n_i}^{max}, \end{aligned} \tag{11}$$

where $\mathfrak{D}$ is valid duty cycle sets of device n_i and $r_{n_i}^{max}$ is the maximum number of packets device n_i could receive. According to (1)-(3), the range of $\mathfrak{D}$ is restricted by the maximum valid SO.

IV. Adaptive Duty Cycle Control

In this section, we first derive the optimal solution of problem $\mathcal{P}_{n_i}$. As the optimal solution is difficult or impractical to implement on computation-limited sensor devices, we further propose a low-complexity RADutyCon, and give its joint-cost upper bound.

A. Optimal Duty Cycle Control

By applying the principle of DP, the problem $\mathcal{P}_{n_i}$ is decomposed into a sequence of subproblems $\mathcal{S}(r_{n_i}^k)$, where $0 \le k \le K$. The objective of each subproblem $\mathcal{S}(r_{n_i}^k)$ is to minimise the sum of joint-cost functions from time period k to K. Thus, the total cost of $\mathcal{P}_{n_i}$ is equal to that of $\mathcal{S}(r_{n_i}^0)$, which means the optimal solution of $\mathcal{S}(r_{n_i}^0)$ is the optimal solution of $\mathcal{P}_{n_i}$. Based on (9), the cost-to-go function $U(r_{n_i}^k)$ of $\mathcal{S}(r_{n_i}^k)$ is

$$\begin{aligned} U(r_{n_i}^k) = \min_{\pi_{n_i} \in \mathfrak{S}} \mathbb{E}\Big\{&\alpha(E_r(r_{n_i}^k) + E_t(f_{n_i}^k)) + H(r_{n_i}^k) \\ &+ \mathbb{E}\{U(r_{n_i}^{k+1})\}\Big\}, \end{aligned} \tag{12}$$

where $H(r_{n_i}^k) = \mathbb{E}\Big\{\alpha E_l(r_{n_i}^k) + \beta D(r_{n_i}^k)\Big\}$ shows the tradeoff between idle listening energy consumption cost and the end-to-end delay cost. For simplicity, we introduce $m_{n_i}^k = q_{n_i}^k + r_{n_i}^k$ and $n_{n_i}^k = f_{n_i}^k - g_{n_i}^k$, combined with (7) and (8), $H(r_{n_i}^k)$ can be rewritten as

$$\begin{aligned} H(m_{n_i}^k) = \mathbb{E}\Bigg\{&\alpha c_l \times \frac{\max(q_{n_i}^{max}, [n_{n_i}^k - m_{n_i}^k]^+)}{q_{max} \times l_{n_i}} \\ &+ \beta c_d \times \frac{\max(q_{n_i}^{max}, [m_{n_i}^k - n_{n_i}^k]^+)}{q_{max} \times l_{n_i}}\Bigg\}. \end{aligned} \tag{13}$$

As the convexity preserved by taking expectation over $n_{n_i}^k$, with each fixed $n_{n_i}^k$, $H(m_{n_i}^k)$ is convex. To take the convexity of $H(m_{n_i}^k)$, we rewrite (12) as

$$U(m_{n_i}^k) = \min_{\pi_{n_i} \in \mathfrak{S}} \mathbb{E}\Bigg\{W(m_{n_i}^k) - \alpha c_r \times \frac{q_{n_i}^k}{q_{max} \times l_{n_i}}\Bigg\}, \tag{14}$$

where

$$\begin{aligned} W(m_{n_i}^k) = &\alpha c_r \times m_{n_i}^k + \alpha E_t(f_{n_i}^k) + H(m_{n_i}^k) \\ &+ \mathbb{E}\Big\{U([m_{n_i}^{k+1} - q_{n_i}^{k+1}]^+))\Big\}. \end{aligned} \tag{15}$$

Then the objective of $\mathcal{S}(r_{n_i}^k)$ is to find the minimum value of (14). Based on (10) and (12)-(15), the optimal duty cycle control at each time period can be found by running DP.

Throrem 1: If $W(m_{n_i}^k)$ is convex, and

$$m_{n_i}^{k\ *} = T_{n_i} = \arg\min_{m_{n_i}^k \in \Re} W(m_{n_i}^k), \tag{16}$$

where $\Re$ as the set of all valid values of $m_{n_i}^k$. Then, the optimal solution of $\mathcal{P}_{n_i}$ is

$$SO_{n_i}^{k\ *} = \begin{cases} \left\lceil \log_2(\frac{r_{n_i}^{k\ *} \times P_s}{b} + D_{bcn}) \right\rceil & \text{if } q_{n_i}(k) < T_{n_i}, \\ \lceil \log_2(D_{bcn}) \rceil & \text{if } q_{n_i}(k) \geq T_{n_i}. \end{cases} \tag{17}$$

Proof: For $k = K$, function $U(m_{n_i}^K)$ is the zero function, so it is convex. Since $c_r < c_d$ and the derivative of $H(m_{n_i}^K)$ tends to $-c_d/l_{n_i}$ as $q_{n_i}^K + r_{n_i}^K \to -\infty$, thus $W(m_{n_i}^{K-1})$ has a derivative that becomes negative as $m_{n_i}^K \to -\infty$ and becomes positive as $m_{n_i}^K \to \infty$. Therefore $W(m_{n_i}^{K-1})$ is convex. As $W(m_{n_i}^{K-1})$ is minimised by T_{n_i}, given the convexity of $U(m_{n_i}^K)$, the convexity of $U(m_{n_i}^{K-1})$ is proved.

For $k = K-2, \cdots, 0$, the above arguments can be repeated: if $U(m_{n_i}^{k+1})$ is convex, we can have $U(m_{n_i}^k)$ and $W(m_{n_i}^k)$ are convex. Substituting (8), $n_{n_i}(k) = f_{n_i}(k) - g_{n_i}(k)$ and $r_{n_i}(k) = m_{n_i}(k) - q_{n_i}(k)$ back into (7), the minimum cost-to-go is attained at $r_{n_i}^{k\ *} = T_{n_i} - q_{n_i}^k$ if $q_{n_i}^k < T_{n_i}$, and at $r_{n_i}^k = 0$ otherwise. ∎

B. Rollout Algorithm Based Duty Cycle Control

Based on the above analysis, the optimal duty cycle of device n_i can be found by running DP. However, DP needs to conduct exhaustive search over all possible solutions at each time period, which is very energy inefficient and time consuming. Thus, it is difficult or impractical for computationally-limited sensor devices to run DP.

Rollout algorithms have demonstrate excellent performance on a variety of dynamic optimisation problems. Interpreted as an approximate DP algorithm, a rollout algorithm estimates the cost-to-go at each time period by estimating future costs while following a heuristic control, referred to as the base policy. The heuristic base control in this paper is inspired by the threshold structure of the optimal control. In order to ensure the stable of the queue length, the device should receive same number of packets as it transmits at each time period. Thus, instead of searching the optimal solution by running DP, the most straight forward approach is to set T_{n_i} equals to the mean value of $f_{n_i}^k$ for each device n_i. Based on (18), the heuristic base control of $\mathcal{P}_{n_i}$ is given as

$$SO_{n_i}^{k} = \begin{cases} \left\lceil \log_2(\frac{r_{n_i}^{k} \times P_s}{b} + D_{bcn}) \right\rceil & \text{if } q_{n_i}(k) < f_{n_i}^k, \\ \lceil \log_2(D_{bcn}) \rceil & \text{if } q_{n_i}(k) \geq f_{n_i}^k. \end{cases} \tag{18}$$

The proposed RADutyCon is the one that attains the minimum of the cost-to-go function

$$U(r_{n_i}^k) = \min_{\pi_{n_i} \in \mathfrak{D}} \Bigg[\mathbb{E}\Bigg\{ \alpha(E_r(r_{n_i}^k) + E_f(f_{n_i}^k) + E_l(r_{n_i}^k)) + \beta D(r_{n_i}^k) + \mathbb{E}\{\tilde{U}(r_{n_i}^{k+1})\} \Bigg\} \Bigg], \tag{19}$$

where $\tilde{U}(r_{n_i}^{k+1})$ is the approximation of $U(r_{n_i}^{k+1})$ based on the heuristic base control.

Given the approximations $\tilde{U}(r_{n_i}^k)$, which is calculated based on the heuristic base control, the computational saving of RADutyCon is evident, as only a single minimisation problem has to be solved at each time period. Noticed that even with readily available approximations $\tilde{U}(r_{n_i}^{k+1})$, the calculation of the minimisation over $\pi_{n_i} \in \mathfrak{D}$ may involve substantial computation. To further save the computation, a subset $\bar{\mathfrak{D}}$ of the promising controls is identified in the proposed RADutyCon. Thus, the minimisation over $\mathfrak{D}$ in (20) is replaced by a minimisation over a subset $\bar{\mathfrak{D}} \subset \mathfrak{D}$.

Throrem 2: Let's denote $\hat{U}(r_{n_i}^k)$ as the estimate cost-to-go of RADutyCon, whose control range is $\bar{\mathfrak{D}} \subset \mathfrak{D}$. $U(r_{n_i}^k)$ as the expected actual cost-to-go incurred by RADutyCon. Then we have $U(r_{n_i}^k) \leq \tilde{U}(r_{n_i}^k)$, which means $\tilde{U}(r_{n_i}^k)$ is the cost-to-go upper bound of RADutyCon.

Proof: For $k = 0, 1, \cdots, K-1$, denote

$$\hat{U}(r_{n_i}^k) = \min_{\pi_{n_i} \in \bar{\mathfrak{D}}} \Bigg[\mathbb{E}\Bigg\{ \alpha(E_r(r_{n_i}^k) + E_f(f_{n_i}^k) + E_l(r_{n_i}^k)) + \beta D(r_{n_i}^k) + \mathbb{E}\{\tilde{U}(r_{n_i}^{k+1})\} \Bigg\} \Bigg]. \tag{20}$$

Thus for all $q_{n_i}^k$, we have $\hat{U}(r_{n_i}^k) \leq \tilde{U}(r_{n_i}^k)$, let

$$\begin{aligned} \hat{U}(r_{n_i}^K) &= G(r_{n_i}^K) \\ &= \alpha(E_r(r_{n_i}^K) + E_f(f_{n_i}^K) + E_l(r_{n_i}^K)) + \beta D(r_{n_i}^K). \end{aligned} \tag{21}$$

Applying backward induction on k, we have $U(r_{n_i}^k) = \hat{U}(r_{n_i}^K) = G(r_{n_i}^K)$ for all $q_{n_i}^K$. Assuming that $\bar{U}(r_{n_i}^{k+1}) \leq \hat{U}(r_{n_i}^{k+1})$ for all $q_{n_i}^{k+1}$, we have

$$\begin{aligned} U(r_{n_i}^k) &= \mathbb{E}\Big\{ G(r_{n_i}^k) + \bar{U}(r_{n_i}^{k+1}) \Big\} \leq \mathbb{E}\Big\{ G(r_{n_i}^k) + \hat{U}(r_{n_i}^{k+1}) \Big\} \\ &\leq \mathbb{E}\Big\{ G(r_{n_i}^k) + \tilde{U}(r_{n_i}^{k+1}) \Big\} = \hat{U}(r_{n_i}^k), \end{aligned} \tag{22}$$

for all $q_{n_i}^k$. The first equality above follows from the definition of the cost-to-go $U(r_{n_i}^k)$ of RADutyCon, while the first inequality follows from the induction hypothesis, and the second inequality follow from the assumption $\hat{U}(r_{n_i}^k) \leq \tilde{U}(r_{n_i}^k)$. Then, we have $U(r_{n_i}^k) \leq \hat{U}(r_{n_i}^k) \leq \tilde{U}(r_{n_i}^k)$ for all $q_{n_i}^k$. Thus, the $\tilde{U}(r_{n_i}^k)$ is a readily obtainable performance upper bound for the cost-to-go function $U(r_{n_i}^k)$. ∎

In addition, two remarks of the proposed RADutyCon are given as follows.

Remark 1: The proposed RADutyCon has lower computation complexity as compared to DP optimal control. If D is the average search range of the devices, the computation complexity of DP algorithm is $O(KD^{N+D})$, while that of the suboptimal control is only $O(KND)$.

Remark 2: The proposed suboptimal controls has lower synchronisation overhead as compared to controls in [7] and [2]. The proposed control does not need additional SYNC packet to ensure the devices are active at the same time as it employs the same BO as defined in IEEE 802.15.4 (2011) and all devices are activated at the beginning of each BI.

V. Simulation Results and Analysis

In this section, the performance of RADutyCon is evaluated in Matlab. We consider a two-hop cluster-tree network as explain in section II. The performance of a benchmark control, DP optimal control, the heuristic base control and RADutyCon will be discussed.

Benchmark control: to reduce the end-to-end delay, the benchmark control aims at maximising the number of received packets $r_{n_i}^k$. The SO is determined based on (9) and the maximum SO is bounded by the service rate of device n_i.

DP optimal control: exhausted search of the optimal $r_{n_i}^{k\ *}$ is processed at each time period based on (18) and $m_{n_i}^k = q_{n_i}^k + r_{n_i}^k$, then the value of the optimal SO^* is determined based on (18).

Heuristic base control: the heuristic base control has a threshold equals to $f_{n_i}^k$. Thus, $r_{n_i}^k = f_{n_i}^k - q_{n_i}^k$, and the value of the SO is determined baed on (19). The maximum SO is bounded by the predefined value $T_{n_i} = f_{n_i}^k$.

RADutyCon: RADutyCon will do one search at each time period to find the minimum value of (20), while the future cost is estimated by applying the heuristic based control. The value of the optimal SO^* is determined based on (10), and the maximum SO is bounded the search range $\bar{\mathfrak{D}}$ at each time period. According *Remark 1*, the search range $\bar{\mathfrak{D}}$ is set to be 15 packets to further reduce the computation complexity.

The performance metrics are average energy consumption per packet, average end-to-end delay and packet drop ratio. The average energy consumption per packet is calculated as the total energy consumption of K time periods over the total number of transmitted packets, and the average end-to-end delay is the total buffered time of the packets over the total number of generated packets in the network. Packet drop ratio is calculated as the number of packets been dropped due to excess the maximum queue length over the total number of the generated packets.

We assume there is no packet loss during the transmission. Packets are dropped when the queue length of the device reaches its maximum (i.e. $q_{n_i}^k - f_{n_i}^k + r_{n_i}^k > q_{n_i}^{max}$). The maximum queue length of FFDs is 50 packets and that of the RFDs is 20 packets. Energy consumption parameters in the simulation are based on CC2420 data sheet [11] and MAC layer parameters are based on IEEE 802.15.4 (2011) standard [3]. The duration of each time period (BI) is 0.49s with $BO = 5$, $f_{n_i}^k$ follows poisson distribution with the mean value equals to 30 packets per active period, and the number of observation time periods K is 100. The results are the averaged values of 1000 runs of the device n_i. Specific simulation parameters are given in TABLE I.

TABLE I
Simulation Parameters

Parameter	Value	Parameter	Value
frequency	2.4 GHz	α	0.2
transmit power	36.5 mw	β	0.4
receive power	41.4 mw	CCA size	8 symbols
idle listen power	41.4 mw	ACK packet size	10 symbols
sleep power	0.042 mw	unit backoff period	20 symbols

Fig. 3 shows the joint-costs of the evaluated control mechanisms. It is shown that the proposed RADutyCon has lower joint-cost as compared to the benchmark control and the base control by the average of 31% and 19.7%, respectively, over the range of evaluated traffic. The joint-cost of RADutyCon is close to that of DP optimal control. Base on *Theorem 2*, the heuristic base control is the joint-cost upper bound for RADutyCon with different search ranges. The improvement of RADutyCon to the heuristic base control is achieved by searching the minimum of the cost-to-go function (20) at each time period. According to *Remark 1*, RADutyCon will be more beneficial when device n_i has large number of child devices, as an exponential reduction of the computation complexity can be achieved.

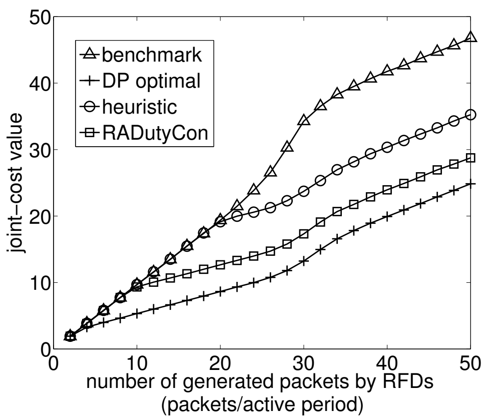

Fig. 3. Rollout algorithm performance bound.

Fig. 4 shows the energy consumption per packet with different arrival rates. The energy consumption curves have a decrease trend along with the increase of generated packets. The change of energy consumption curve of RADutyCon between 8 packets/active period and 15 packets/active period is because the radical increase of SO, which leads to higher idle listening energy consumption. The proposed RADutyCon achieves lower energy consumption compared to benchmark control and the heuristic base control after 12 packets/active period. After 30 packets/active period, the number of transmitted packets is relevant stable, thus the energy consumption curves keep flat for all examined controls.

The end-to-end delay curves in Fig. 5 have same trend with the results in [2]. End-to-end delay of RADutyCon is lower than that of DP optimal. The end-to-end delay curve of RADutyCon begins to decrease after 20 packets per/active period. This is due to the fact that a packet can only be sent out once the existing buffered packets are cleared. With more generated packets by RFDs, the increased number of dropped packets reduces the number of buffered packets which are generated in earlier time periods. Thus, the buffer time is shortened for the packets generated in later time periods, thereby the averaged end-to-end delay is decreased. Compared with Fig. 4, it is clear that the decrease of energy consumption is at the cost of increasing end-to-end delay.

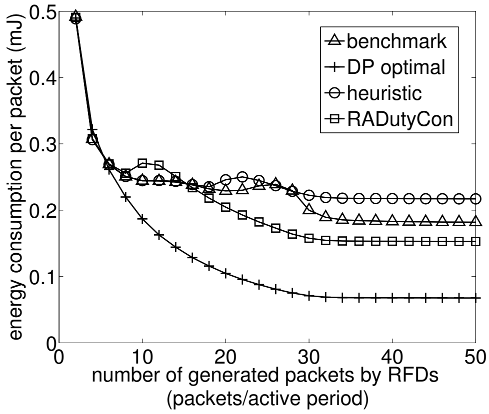

Fig. 4. Energy consumption with different arrival rates.

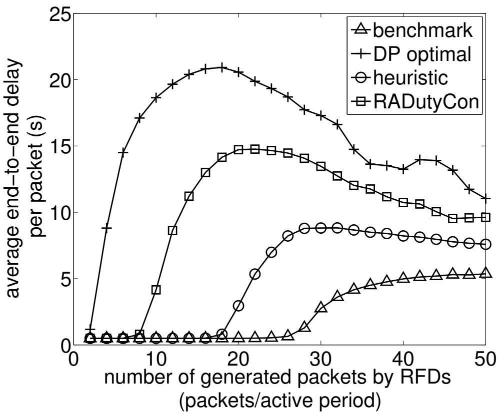

Fig. 5. End-to-end delay with different arrival rates.

Fig. 6 shows the packet drop ratio of the evaluated control mechanisms. The packet drop ratio of RADutyCon has close performance compared with that of the heuristic base control and DP optimal control. The higher packet drop ratio than that of the benchmark control is because the reduced active periods of RADutyCon increases the number of buffered packets. Hence, the possibility of packet drop is increased due to limited maximum queue length.

VI. CONCLUSION

In this paper, we derived the optimal duty cycle control to minimise the expected joint-cost of energy consumption and end-to-end delay for 802.15.4 based WSANs. To reduce the

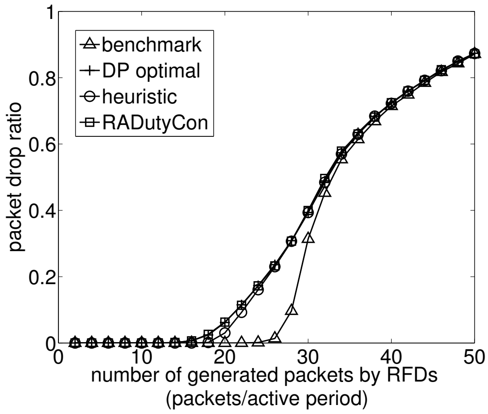

Fig. 6. Packet drop ratio with different arrival rates.

computation complexity, RADutyCon is proposed. Simulation results shown that RADutyCon can effectively reduces the joint-cost of energy consumption and end-to-end delay under various network traffic. RADutyCon achieved lower joint-costs over the benchmark control and the heuristic base control by the average of 31% and 19.7%, respectively, over the range of evaluated traffic. The joint-cost is similar to that of DP optimal control. In addition, an exponential reduction of the computation complexity is achieved by RADutyCon.

REFERENCES

[1] M. R. Palattella, N. Accettura, M. Dohler, L. A. Grieco, and G. Boggia, ”Traffic Aware Scheduling Algorithm for Reliable Low-Power Multi- Hop IEEE 802.15.4e Networks,” in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC*, September 2012.

[2] H. Byun and J. Yu, ”Adaptive Duty Cycle Control with Queue Management in Wireless Sensor Networks,” *IEEE Transactions on Mobile Computing*, vol. 12, no. 6, pp. 1214-1224, June, 2013

[3] *IEEE std. 802.15.4, Part. 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE Std., 2011.

[4] P. Lin, C. Qiao, and X. Wang, ”Medium Access Control With A Dynamic Duty Cycle For Sensor Networks,” in *Proc. IEEE Wireless Communications and Networking Conference, WCNC*, vol. 3. Atlanta, GA, USA, March 2004.

[5] S. H. Yang, H. W. Tseng, E. Wu, and G. H. Chen, ” Utilization Based Duty Cycle Tuning MAC Protocol for Wireless Sensor Networks, ” in *Proc. IEEE Global Telecommunications, GLOBECOM*. Dallas, Texas, November 2005. pp. 3258 - 3262.

[6] Z. Yuqun, F. Chen-Hsiang, I. Demirkol, and W.B. Heinzelman, ”Energy-Efficient Duty Cycle Assignment for Receiver-Based Convergecast in Wireless Sensor Networks,” in *Proc. IEEE Global Telecommunications Conference, GLOBECOM*, pp. 1-5, December 2010.

[7] X. Wang, X. Wang, G. Xing, and Y. Yao, ”Dynamic Duty Cycle Control for End-to-End Delay Guarantees in Wireless Sensor Networks,” in *Proc. IEEE International Workshop on Quality of Service, IWQoS*, Beijing, China, June 2010.

[8] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd Edition, Athena Scientific, 2005.

[9] T.R. Park, T.H. Kim, J.Y. Choi, S. Choi and W.H. Kwon, ”Throughput and energy consumption analysis of IEEE 802.15.4 slotted CSMA/CA,” *Electronics Letters*, vol 41, no 18, 2005.

[10] H. Scarf, ”The Optimality of (s,S) Policies for the Dynamic Inventory Problem,” in *Proc. of the 1st Stanford Symposium on Mathematical Methods in the Social Sciences*, Stanford, CA, 1960.

[11] CC2420 2.4 GHz IEEE 802.15.4 Compliant and ZigBee Ready RF Transceiver, online: http://www.ti.com/product/cc2420