A Markovian Approach to Modeling the Optimal Lifetime of Multi-hop Wireless Sensor Networks

Jian Lin and Mary Ann Weitnauer*
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, Georgia 30332–0250
Email: jlin61@gatech.edu; maweit@gatech.edu

***Abstract*—A Markov Decision Process (MDP) framework is presented for modeling the lifetime of Multi-hop Wireless Sensor Networks (WSNs). The model applies to both non-cooperative and cooperative (CT) networks. To our knowledge, this is the first work to model the *lifetime* of multi-hop networks that jointly considers the dynamics of MAC layer link admission, routing layer queuing and energy evolution. We propose a new algorithm that exploits the Stochastic Shortest Path (SSP) structure and Mixed Integer Linear Programming (MILP) to efficiently solve the problem. Numerical results on the optimal lifetime of non-CT and CT networks are presented to validate the model.**

I. Introduction

Wireless Sensor Networks (WSNs) usually consist of a large number of low cost, low capability and battery-powered sensors (nodes). Typical event reporting and monitoring applications present *multi-hop* transmission scenarios in which nodes forward local measures through multiple Single-Input-Single-Output (SISO) hops to the remote central processors (Sinks). Therefore, it is highly desirable that the networks can sustain long before some nodes become energy-exhausted. An unresolved challenge is to derive the optimal *lifetime* for a general multi-hop wireless sensor network under general routing layer and MAC layer constraints, which can serve as a benchmark for evaluating performance of practical and heuristic routing and MAC protocols[1]. In this study, we present a Markov Decision Process (MDP) framework for modeling the lifetime of multi-hop WSNs. To our knowledge, this is the first work to model the optimal *multi-hop* WSN lifetime by jointly considering MAC layer link admission, routing queuing and energy evolution.

The lifetime is usually defined as the number of packets delivered upon the first node death or a portion of nodes' death. Therefore, it is limited by the *energy holes* resulted from bottleneck nodes that are near the Sinks, because these nodes consume energies faster than others [1]. Though many existing protocols aim at minimizing energy consumption, they do not solve the energy hole problem. Cooperative transmission (CT) range extension has been recently shown to correct the energy imbalance and extend the lifetime [2] through cooperative routing forwarding, assuming perfect link scheduling. Our previous works also addressed the multi-flow contention in CT network through scheduling MAC protocols [3] [4]. However, the performance gap between these heuristic protocols and the optimal lifetime under constraints remains unresolved.

Theoretical work to derive the optimal lifetime for non-CT networks includes the linear programming (LP) approach [5], which was recently extended to analyze CT networks [6] by considering all possible virtual Multiple-Input-Single-Output (VMISO, a.k.a. CT) links along with SISO links. The LP formulation requires traffic balance conditions and thus ignores packet losses and retransmissions. Further, the LP formulation cannot capture link scheduling with MAC layer constraints, i.e., the LP formulation and the proposed routing protocols in [5] [6] assume link transmissions are perfectly scheduled so that there are no link failures from collisions.

To capture the dynamics of energy evolution, the Markov Decision Process (MDP) has been used to formulate the *single-hop* network lifetime in [7]. It is shown that the MDP is equivalent to a Stochastic Shortest Path problem. The authors consider sensor scheduling in a scenario where only a fraction of sensors collect information and communicate directly with the Sink that is one-hop away. The model in [7] is somehow *incomplete*, because it does not consider MAC constraints and does not apply to multi-hop WSNs where the sensed information needs to be forwarded through multiple hops to the Sink(s); also, it only applies to non-CT networks. Thus, the applications of the analytical approach in [7] are limited.

Our contributions in this paper include a unified Markov Decision Process (MDP) framework to model the lifetime of *multi-hop* WSNs. This framework applies to both non-cooperative and cooperative networks. Departing significantly from analysis for single-hop networks, the MDP in this paper is *complete* in the sense that we jointly consider the MAC constraints, the packet transfers between queues, and the energy evolution. A new algorithm is proposed to solve the lifetime, which exploits the Stochastic Shortest Path (SSP) structure of the problem and Mixed Integer Linear Programming (MILP).

The rest of this paper is structured as follows. Section II describes the system model and the assumptions. In Section III, We formulate the *multi-hop* WSN lifetime as a Markov Decision Process problem. Section IV describes our proposed solution algorithm. Section V provides numerical results for non-CT networks and CT networks. The concluding remarks are given in Section VI.

*Mary Ann Weitnauer was formerly Mary Ann Ingram. The authors gratefully acknowledge support for this research from the National Science Foundation under grant CNS-1017984.

[1]This paper does not present a new protocol, but rather a method for calculating the optimal lifetime.

978-0-7695-5124-1/13 $31.00 © 2013 IEEE
DOI 10.1109/MILCOM.2013.288

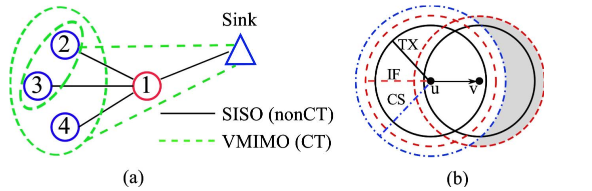

Figure 1: (a) A network example. Node 2 can form a VMISO link to the Sink by cooperating with (i) $\mathcal{H}(2) = \{\text{Node } 3\}$ or (ii) $\mathcal{H}(2) = \{\text{Node } 3, \text{ Node } 4\}$; (b) An illustration of interference model. Node v is the receiver of Node u. Node u's hidden nodes in the gray area will interfere v's reception.

II. Problem Statement

A. System Model and Assumptions

1) Network Model: We consider a general multi-hop WSN whose topology can be modeled as a directed graph $G = (V, E)$, where V is the set of nodes. $V = \{1, \cdots, N\}$. The Sink's ID is 0. Let V^* denote $\{0, 1, \cdots, N\}$. Each Node i (except the Sink) has a set of cooperator groups:

$$\mathbf{H}(i) = \{\mathcal{H}(i)\}, \ i \in V. \tag{1}$$

where $\mathcal{H}(i)$ is a candidate cooperator group (as in Fig.1(a)), and $E = E^{so} \cup E^{vo}$ denotes the set of links whose end nodes are within transmission range. Specifically, E^{so} includes the links formed by SISO transmission, and E^{vo} includes the long-haul links formed by VMISO transmission.

$$E^{so} = \{(i, j, \varnothing) : i, j \in V\}, \tag{2}$$

$$E^{vo} = \{(i, 0, \mathcal{H}(i)) : i \in V, \mathcal{H}(i) \in \mathbf{H}(i)\}. \tag{3}$$

A SISO link $(i, j, \varnothing)$ exists if Transmitter i and Receiver j are within direct transmission range. A VMISO link $(i, 0, \mathcal{H}(i))$ means that Sink 0 can be reached by the source node i with its cooperators $\mathcal{H}(i)$. For a link $l \in E$, we denote $s(l)$ as the source of the link and $d(l)$ as the destination of the link.

2) Interference Model: The interference range (IF) of a link $l \in E$ is denoted as $R(l)$. Note that link $(i, j, \cdot)$ interferes with link $(u, v, \cdot)$ if either the distance $dist(i, v) \leq R(i, j, \cdot)$ or $i = v$, similar to [8] . For instance, in Fig. 1(b), the Node u and any node in the gray area can start transmission at the same time because they are out of carrier sensing (CS) range of (i.e., hidden from) each other. Also, located in Node v's interference (IF) range, the hidden node's transmission interferes Node v's reception. This phenomenon directly results from the CSMA mechanism of the MAC layer and the relationship between the three ranges (TX, IF, CS) [9][2].

3) MAC Assumptions: Pertaining to the CSMA of MAC, we make the following assumptions, similar to [10]:

(a.1) A node cannot receive and transmit at the same time.

(a.2) A node can transmit if none of its neighbors (in CS range) is transmitting.

[2]The CTS in CSMA/CA cannot eliminate hidden terminals, because RTS packets can still collide [9].

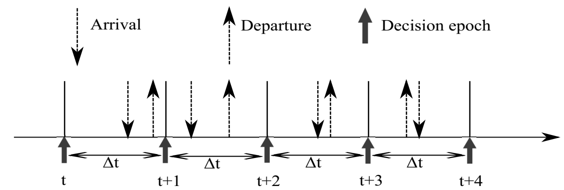

Figure 2: A timeline illustration of the decision process model for the network.

Table I: Parameters of the MDP model

$\Psi(\mathcal{L})$	Collision-free links in the transmission set $\mathcal{L}$
E^{so}, E^{vo}	non-CT and CT links in transmission set $\mathcal{L}$
β^l	Probability that link l finishes transmission
$\alpha(i)$	Exogenous arrival rate of Node i (destined to the Sink 0)
q^{max}, e^{max}	The buffer size, battery capacity of nodes
e_{tx}, e_{rx}	Energy consumption in SISO (source and destination)
$e^{CT}_{init}, e^{CT}_{co}$	Energy consumption in VMISO (source and cooperators)
e_{th}	Threshold of residual energy in lifetime definition

(a.3) Link errors result only from collisions due to hidden terminals.[3]

(a.4) Nodes receive with perfect capture, i.e., a packet will be successfully decoded if the receiver and all its neighbors are not transmitting at the start of packet.

We consider a mini-time-slotted system where slots are normalized to integral units $t \in \{1, 2, 3, \cdots\}$. By time slot t, we mean the time duration in between $[t, t+1]$. Let $\alpha^d(i)$ represent the exogenous packet arrival rate to Node i of *commodity* d (destined to Sink d), which is assumed *i.i.d.* over time slots.

B. Energy Model and Lifetime Definition

Let $\mathbf{e} = (e_1, \cdots, e_N)$ denote the network energy profile at the transmission epoch (which is also the decision epoch). Note that the residual energy of the Node i, e_i, is a random variable conditioned on previous states and decision rules. A node is said to become dead when its energy drops to a predefined threshold e_{th}. The definitions of network lifetime depend on the applications. In this paper, we define the lifetime as the number of packets successfully delivered to the Sink when the first node dies.

III. Markov Decision Process Formulation

Consider a multi-hop WSN consisting of N sensor nodes each with an initial energy e^{max}. We model the state evolution of the network by an MDP, a controlled Markovian dynamic system. The control model is expressed by the 4-tuple: $\{S, \mathcal{A}(S), q(\mathbf{s}' \mid \mathbf{s}, a), g(\mathbf{s}, a)\}$ (state space, action space, transition kernel, rewards) [11]. A timeline illustration of the MDP process is given in Fig.2.

1) Traffic Model: i). We assume that the packet duration of a link l is exponentially distributed with the expectation T^l_{mean}. This assumption is used by several other authors [10] [12];

[3]Our model can be extended to also consider link errors due to channel fading.

though inaccurate, it is necessary to make the MDP problem tractable. Due to the memoryless property of the exponential distribution, given that a packet is being transmitted at the beginning of a time slot, it completes within the slot of length Δt with probability $\beta^l(\Delta t) = 1 - e^{-\Delta t / T^l_{mean}}$, regardless of the number of time slots it has already been transmitted. The slot duration Δt is small enough so that we can assume at most one link can finish transmission during a specific time slot $[t, t + \Delta t], \forall t$, i.e., the probability that more than one link finishes transmission is $o(\Delta t)$. ii). It is also assumed that during one slot, at most one node can self-generate a packet. The probability that only Node i has a new self-generated packet is thus $\alpha(i) \prod_{j \in V, j \neq i} (1 - \alpha(j))$, where $\alpha(i)$ is the probability that Node i has a new packet, which is determined by the application. Under uniform exogenous traffic, we have $\alpha(1 - \alpha)^{N-1}, \forall i \in V$. The probability that more than one nodes self-generate packets tends to $o(\Delta t)$.

2) Network State Space: The evolution of network states can be modeled by an embedded Markov chain $\{S(t), t \geq 0\}$. We define the network state at the beginning of time slot t as $S(t) \triangleq (\mathcal{L}(t), \mathbf{q}(t), \mathbf{e}(t))$. The state comprises three component processes:

2.1) The transmission set $\mathcal{L}(t) = \mathcal{L}^{so}(t) \cup \mathcal{L}^{vo}(t)$ of links that are active (in transmission) at time t. Denote the space as $\mathfrak{S} = \{\mathcal{L}(t)\}$. $\mathcal{L}^{so}(t)$ and $\mathcal{L}^{vo}(t)$ denote the SISO links and the VMISO links, respectively. This component includes the collisions in the MAC layer. We denote the links that are free of collision as $\Psi(\mathcal{L}) \subset \mathcal{L}(t)$. $\Psi(\mathcal{L})$ is deterministic given $\mathcal{L}(t)$.

Given a graph $G = (V, E)$, a matching $M(G)$ in G is a set of non-adjacent edges (i.e., no two edges share a common vertex). Therefore, $\mathcal{L}(t) \in \{M(G)\}, \forall t \geq 0$. In addition, for any two links $l_1, l_2 \in \mathcal{L}(t)$, the source nodes of the two links $(s(l_1), s(l_2))$ are out of carrier sensing (CS) range of each other, due to CSMA. Therefore, the state space of $\mathcal{L}(t)$ is determined by the graph G and the carrier sensing matrix $\mathbf{H} = [h_{ij}]_{i,j \in V}$ of the network, where the element $h_{i,j} = 1$ if and only if (*iff*) Node i and Node j are within CS range of each other. Finding all the matchings in a graph is *NP-complete*, but despite its hardness many algorithms for finding them have been studied [13].

2.2) The queue length (vector) $\mathbf{q}(t) \in \{0, 1, \cdots, q^{max}\}^N$ of each node, where q^{max} is the buffer size. $\mathbf{q}(t)$ monitors the congestion in the network. The queue backlog contains the packets that arrived both exogenously from the sensing application and endogenously from other nodes.

2.3) The residual energy (vector) $\mathbf{e}(t)$ of each node. $\mathbf{e}(t)$ determines when the network reaches a termination state (i.e., first node death).

We will show that each component process evolves as a controlled Markov process, and therefore the resulting system state evolution is a controlled Markov process.

3) Action Space: In the beginning of each time slot, t, new link(s) $(a(t))$ may join the remaining *transmission set* $\mathcal{L}(t)$, which comprises those links that did not complete transmission during Slot $t-1$. Note that $\mathcal{L}(t) \subset \mathcal{L}(t-1) \cup a(t-1)$. Again because Δt is small we assume at most one link can join $\mathcal{L}(t)$. Note that link l can join $\mathcal{L}(t)$ iff its source has a non-empty queue and $\mathcal{L}(t) \cup \{l\}$ is also a transmission set. Thus, the candidate links are denoted as

$$\mathcal{A}(S(t)) = \left\{ l \in E : \mathbf{q}_{s(l)}(t) > 0, l \notin \mathcal{L}(t), \mathcal{L}(t) \cup \{l\} \subset \mathfrak{S} \right\} \tag{4}$$

$\mathcal{A}(S(t))$ represents the action space at time t when the system state is $S(t)$. Note that the null set $\varnothing \in \mathcal{A}(S(t))$, and therefore the action (*link*) can be either a "CT" link, a "non-CT" link or null (no new transmission).

4) Controlled Markovian Dynamics:

4.1) *Transmission set dynamics*: Let $a(t) \in \mathcal{A}(S(t))$ denote the link selected for joining the transmission set. Let $P\left\{z^l(t) \mid \mathcal{L}(t), a(t)\right\}$ denote the probability that only link $l \in \mathcal{L}(t) \cup a(t)$ finishes transmission during Slot t. Let $P\{z^{\varnothing}(t) \mid \mathcal{L}(t), a(t)\}$ denote the probability that no link finishes transmission during Slot t. The time index is dropped when there is no ambiguity. For abbreviation, we denote the transition kernel as $P\{\mathcal{L}' \mid \mathcal{L}, a\} := P\{\mathcal{L}(t+1) \mid \mathcal{L}(t), a(t)\}$. Let $\Delta\mathcal{L} \triangleq \mathcal{L} \cup \{a\} \setminus \mathcal{L}'$, then we have:

$$P\{\mathcal{L}' \mid \mathcal{L}, a\} = \begin{cases} P\left\{z^l \mid \mathcal{L}, a\right\} & \text{if } \Delta\mathcal{L} = l, \\ P\left\{z^{\varnothing} \mid \mathcal{L}, a\right\} & \text{if } \Delta\mathcal{L} = \varnothing, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

where,

$$P\left\{z^l \mid \mathcal{L}, a\right\} = \beta^l \prod_{\substack{k \in \mathcal{L} \cup \{a\} \\ k \neq l}} \left(1 - \beta^k\right), \forall l \in \mathcal{L} \cup \{a\}. \tag{6}$$

$$P\left\{z^{\varnothing} \mid \mathcal{L}, a\right\} = \begin{cases} \prod_{l \in \mathcal{L} \cup \{a\}} \left(1 - \beta^l\right) & \text{if } \mathcal{L} \cup \{a\} \neq \varnothing, \\ 1 & \text{if } \mathcal{L} \cup \{a\} = \varnothing. \end{cases} \tag{7}$$

4.2) *Queue length dynamics*: The queue length distribution reflects the loads across the network and directly relates to the first node death. The queue length evolves according to the traffic balance equation in vector form:

$$\mathbf{q}'(t) \triangleq \mathbf{q}(t+1) = \mathbf{q}(t) + \mathbf{R}\mathbf{M}(t)\mathbf{v}(t) + \mathbf{f}(t), \tag{8}$$

where $\mathbf{M}(t)$ is a $|E| \times |E|$ diagonal matrix, whose diagonal element $\mathbf{M}_i(t) = 1$ if the link i's transmission is *completed* and *successful*[4], and $\mathbf{M}_i(t) = 0$ otherwise (links are numbered to be indexable). Note that under our assumption, at most one diagonal element can be 1, thus $\sum_{i \in E} \mathbf{M}_i(t) \leq 1$. The transmission schedule vector $\mathbf{v}$ is determined by the system state at time slot t, satisfying $\mathbf{v}_i = 1, \forall i \in \mathcal{L}(t) \cup a(t)$ and $\mathbf{v}_i = 0$ otherwise. Matrix $\mathbf{R}$ is the $N \times |E|$ routing matrix. The element of $\mathbf{R}$ in its ith row and lth column is

$$r_{il} = \begin{cases} 1 & \text{if } d(l) = i \text{ and Node } i \text{ is not the sink}, \\ -1 & \text{if } s(l) = i, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

[4]A finished transmission is not necessarily successfully decoded by the receiver, because of collision due to hidden terminals; in case of failure, the packet remains in the queue of the transmitter.

$\mathbf{f}(t)$ is a vector with with its ith element $\mathbf{f}_i(t)$ being the number of exogenous packets arriving at Node i during Slot t. Under the assumption that at most one node self-generates a packet, it follows $\sum_{i=1}^{N} \mathbf{f}_i(t) \leq 1$. Let $\mathbf{I}_i$ represent a vector with its ith element being 1 and other elements being 0, $1 \leq i \leq N$, and let $\mathbf{I}_0$ be the zero vector. The probability that only Node i or no node ($i = 0$) self-generates a packet is given by:

$$P\{\mathbf{I}_i\} = \begin{cases} \alpha(i) \prod_{j \in V, j \neq i} (1 - \alpha(j)) & \text{if } i \in V, \\ \prod_{j \in V} (1 - \alpha(j)) & \text{if } i = 0. \end{cases} \tag{10}$$

Case 1 (c1): $\mathbf{q}' = \mathbf{q}$. This can happen for two reasons: (*c1:1*) no link transmission affects the queue state and no new exogenous arrival affects the queues; and (*c1:2*) some link $(i, 0, \cdot)$ destined to the Sink successfully transmits and the source self-generates a new packet.

$$\begin{aligned} P_{c1:1} = & \Big(\mathbb{1}\{\Delta\mathcal{L} = \varnothing\} + \mathbb{1}\{\Delta\mathcal{L} \in \overline{\Psi}(\mathcal{L} \cup a)\} \\ & + \mathbb{1}\{\Delta\mathcal{L} \in \Psi(\mathcal{L} \cup a), q_{d(\Delta\mathcal{L})} = q^{max}\} \Big) \\ & \cdot \Big(P\{\mathbf{I}_0\} + \sum_{q_k = q^{max}, k \in V} p\{\mathbf{I}_k\} \Big) \end{aligned} \tag{11}$$

$$\begin{aligned} P_{c1:2} = & \mathbb{1}\{\Delta\mathcal{L} \in \Psi(\mathcal{L} \cup a), s(\Delta\mathcal{L}) = i, d(\Delta\mathcal{L}) = 0\} \\ & \cdot P\{\mathbf{I}_i\}, \end{aligned} \tag{12}$$

where, $\mathbb{1}(\cdot)$ is an indicator function.

Case 2 (c2): $\mathbf{q}' = \mathbf{q} + \mathbf{I}_j, j \in V$. This case includes two possibilities: (*c2:1*) some SISO link $(\cdot, j, \varnothing)$that is directed to Node j successfully transmits and the source self-generates a new packet; and (*c2:2*) no link transmission affects the queue state and there is a new exogenous packet arrival to Node j.

$$P_{c2:1}(j) = \mathbb{1}\{\Delta\mathcal{L} \in \Psi(\mathcal{L} \cup a), d(\Delta\mathcal{L}) = j\} P\{\mathbf{I}_{s(\Delta\mathcal{L})}\}. \tag{13}$$

$$\begin{aligned} P_{c2:2}(j) = & \Big(\mathbb{1}\{\Delta\mathcal{L} = \varnothing\} + \mathbb{1}\{\Delta\mathcal{L} \in \overline{\Psi}(\mathcal{L} \cup a)\} \\ & + \mathbb{1}\{\Delta\mathcal{L} \in \Psi(\mathcal{L} \cup a), q_{d(\Delta\mathcal{L})} = q^{max}\} \Big) \\ & \cdot P\{\mathbf{I}_j\}. \end{aligned} \tag{14}$$

Case 3 (c3): $\mathbf{q}' = \mathbf{q} - \mathbf{I}_i, i \in V$. This happens because some link $(i, 0, \cdot)$ destined to the Sink successfully transmits and no new exogenous arrival affects the queues.

$$\begin{aligned} P_{c3}(i) = & \mathbb{1}\{\Delta\mathcal{L} \in \Psi(\mathcal{L} \cup a), \Delta\mathcal{L} = (i, 0, \cdot)\} \\ & \cdot \Big(P\{\mathbf{I}_0\} + \sum_{q'_k = q^{max}, k \in V} P\{\mathbf{I}_k\} \Big). \end{aligned} \tag{15}$$

Case 4 (c4): $\mathbf{q}' = \mathbf{q} - \mathbf{I}_i + \mathbf{I}_j, i, j \in V, i \neq j$. This case includes two possibilities: (*c4:1*) Some SISO link $(i, j, \varnothing)$ successfully transmits and no change of queue due to new exogenous arrival; and (*c4:2*) link $(i, 0, \cdot)$ successfully transmits and Node j self-generates a new packet.

$$\begin{aligned} P_{c4:1}(i, j) = & \mathbb{1}\{\Delta\mathcal{L} \in \Psi(\mathcal{L} \cup a), \Delta\mathcal{L} = (i, j, \varnothing)\} \\ & \cdot \Big(P\{\mathbf{I}_0\} + \sum_{q'_k = q^{max}, k \in V} P\{\mathbf{I}_k\} \Big). \end{aligned} \tag{16}$$

$$P_{c4:2}(i, j) = \mathbb{1}\{\Delta\mathcal{L} \in \Psi(\mathcal{L} \cup a), \Delta\mathcal{L} = (i, 0, \cdot)\} P\{\mathbf{I}_j\}. \tag{17}$$

Case 5 (c5): $\mathbf{q}' = \mathbf{q} - \mathbf{I}_i + 2\mathbf{I}_j, i, j \in V, i \neq j$. This occurs because the SISO link $(i, j, \varnothing)$ successfully transmits and Node j self-generates a new packet.

$$P_{c5}(i, j) = \mathbb{1}\{\Delta\mathcal{L} \in \Psi(\mathcal{L} \cup a), \Delta\mathcal{L} = (i, j, \varnothing)\} \cdot P\{\mathbf{I}_j\}. \tag{18}$$

Case 6 (c6): $\mathbf{q}' = \mathbf{q} - \mathbf{I}_i + \mathbf{I}_j + \mathbf{I}_k, i, j, k \in V, i \neq j \neq k$. This occurs because the SISO link $(i, j, \varnothing)$ successfully transmits and a different Node k self-generates a new packet.

$$\begin{aligned} P_{c6}(i, j, k) = & \mathbb{1}\{\Delta\mathcal{L} \in \Psi(\mathcal{L} \cup a), \Delta\mathcal{L} = (i, j, \varnothing)\} P\{\mathbf{I}_k\} \\ & + \mathbb{1}\{\Delta\mathcal{L} \in \Psi(\mathcal{L} \cup a), \Delta\mathcal{L} = (i, k, \varnothing)\} P\{\mathbf{I}_j\}. \end{aligned} \tag{19}$$

Therefore, the transition kernel of $\mathbf{q}(t)$ is expressed as Eq.(20).

4.3) *Energy evolution dynamics*: The process $\mathbf{e}(t)$ is dictated by transmission energy and receiving energy consumption incurred during a finished link transmission, regardless of success or failure. In the CT case, the cooperators consume additional energy in receiving the broadcast packet initiated by the source and in conducting CT. The transition kernel of $\mathbf{e}(t)$ is given in Eq.(21).

Given the transition kernel of the component processes, the whole system's transition is derived in Eq.(22)-(24).

5) Expected Total Rewards: During a time Slot, the system obtains a reward $g(\mathbf{s}, a) = 1$ if a packet was delivered to the Sink, and $g(\mathbf{s}, a) = 0$ otherwise. Then, for $\mathbf{s} \in S \setminus S_t$ (S_t are termination states), we have:

$$g(\mathbf{s}, a) \triangleq E[r] \tag{25}$$

$$= \sum_{\substack{l \in \Psi(\mathcal{L} \cup a) \\ d(l) = 0}} P\{z^l \mid \mathcal{L}, a\} \mathbb{1}\{\Psi(\mathcal{L} \cup a) \neq \varnothing\}. \tag{26}$$

And for $\mathbf{s} \in S_t$, $g(\mathbf{s}, a) = 0$. The expected total rewards of the process starting from an initial state $\mathbf{s}$, under the policy $\boldsymbol{\pi}$ is denoted by

$$\mathcal{J}^{\boldsymbol{\pi}}(\mathbf{s}) = \sum_{t=1}^{\infty} g(\mathbf{s}(t), a(t)). \tag{27}$$

6) MDP formulation. A transmission policy is a series of decision rules $\boldsymbol{\pi} = [a(1), a(2), \cdots]$, where $a(t) : \mathcal{S}(t) \rightarrow \mathcal{A}(\mathcal{S}(t))$. The maximum lifetime $\mathcal{J}^*(\mathbf{s})$ is given by:

$$\mathcal{J}^*(\mathbf{s}) = \max_{\boldsymbol{\pi}} \mathcal{J}^{\boldsymbol{\pi}}(\mathbf{s}). \tag{28}$$

The optimal lifetime $\mathcal{J}^*(\mathbf{s})$ is the unique solution of the Bellman's optimality equation [11]:

$$\mathcal{J}(\mathbf{s}) = \max_a \Big\{ g(\mathbf{s}, a) + \sum_{\mathbf{s}'} P\{\mathbf{s}' \mid \mathbf{s}, a\} \ \mathcal{J}(\mathbf{s}') \Big\}. \tag{29}$$

A policy $\boldsymbol{\pi}^*$ is optimal if it achieves the maximum expected lifetime for all starting state, i.e.,

$$\mathcal{J}^{\pi^*}(\mathbf{s}) = \mathcal{J}^*(\mathbf{s}), \forall \mathbf{s} \in \mathcal{S} \setminus \mathcal{S}_t. \tag{30}$$

IV. Computational Method

Consider that the total energies are non-increasing, we group the states according to the sum energy (in decreasing order), and solve the stages backwards. The transmission sets are computed using the Bron-Kerbosch algorithm [13], which is widely used and referred to as one of the fastest. In each stage, the computation of the energy and queue state spaces is similar to the *Bin-Ball* problem[5]. For instance, the number of ways to allocate n_1 units of energies into n_2 nodes with minimum energy requirement of $\theta = e_{th} + 1$ (in $\mathcal{S} \setminus \mathcal{S}_t$), is given by:

$$\sum_{l=0}^{n_2} (-1)^l \binom{n_2}{l} \binom{n_2 + n_1 - n_2\theta - l(e^{max} + 1 - \theta) - 1}{n_2 - 1}. \tag{31}$$

For an integer m (stage), the set of states S_m is defined as:

$$S_m = \left\{ (\mathcal{L}, \mathbf{e}, \mathbf{q}) \mid \sum_{i \in V} \mathbf{e}_i = m \right\},\ m = N\theta, \cdots, Ne^{max}. \tag{32}$$

[5]The *Bin-Ball* problem refers to enumerating the ways of allocating n_1 balls into n_2 bins. In our problem, the "bins" are the nodes, the "balls" are the total energies or queued packets.

For each S_m and $\mathbf{s} \in S_m$, we have

$$\mathcal{J}^*(\mathbf{s}) = \max_{a \in \mathcal{A}(\mathbf{s})} \left\{ g(\mathbf{s}, a) + \sum_{\mathbf{s}'} P\{\mathbf{s}' \mid \mathbf{s}, a\} \mathcal{J}^*(\mathbf{s}') \right\} \tag{33}$$

$$= \max_{a \in \mathcal{A}(\mathbf{s})} \left\{ g(\mathbf{s}, a) + \sum_{\mathbf{s}' \in S_{N\theta} \cup \cdots \cup S_{m-1}} P\{\mathbf{s}' \mid \mathbf{s}, a\} \mathcal{J}^*(\mathbf{s}') + \sum_{\mathbf{s}' \in S_m} P\{\mathbf{s}' \mid \mathbf{s}, a\} \mathcal{J}^*(\mathbf{s}') \right\},\ \mathbf{s} \in S_m. \tag{34}$$

Therefore, we formulate a Mixed Integer Linear Programming (MILP) problem (Subproblem-1):

$$\text{Minimize} \quad \sum_{\mathbf{s} \in S_m} \mathcal{J}(\mathbf{s}) \tag{35}$$

$$\text{S.t.} \quad \mathcal{J}(\mathbf{s}) - \sum_{\mathbf{s}' \in S_m} P(\mathbf{s}' \mid \mathbf{s}, a) \mathcal{J}(\mathbf{s}') \geq b(\mathbf{s}, a) \tag{36}$$

$$\mathcal{J}(\mathbf{s}) \in Z^+, \mathbf{s} \in S_m. \tag{37}$$

where, $b(\mathbf{s}, a)$ are obtained from previously computed stages:

$$b(\mathbf{s}, a) = g(\mathbf{s}, a) + \sum_{\mathbf{s}' \in S_{N\theta} \cup \cdots \cup S_{m-1}} P\{\mathbf{s}' \mid \mathbf{s}, a\} \mathcal{J}^*(\mathbf{s}'). \tag{38}$$

The computational method is summarized in Alg.1.

V. Numerical Results

We present some numerical results on the optimal lifetime of the *funnel* topology network as in Fig. 1(a). Besides the computational limitation inherent to MDP, the motivations for choosing a small network are as follows. First, the observation from our previous work [3] shows that under duty-cycling, a

$$P\{\mathbf{q}' \mid \mathcal{L}', \mathcal{L}, \mathbf{q}, a\} = \begin{cases} P_{c1:1} + P_{c1:2} & \text{if } \mathbf{q}' = \mathbf{q}, \\ P_{c2:1}(j) + P_{c2:2}(j) & \text{if } \mathbf{q}' = \mathbf{q} + \mathbf{I}_j, j \in V, \\ P_{c3}(i) & \text{if } \mathbf{q}' = \mathbf{q} - \mathbf{I}_i, i \in V, \\ P_{c4:1}(i,j) + P_{c4:2}(i,j) & \text{if } \mathbf{q}' = \mathbf{q} - \mathbf{I}_i + \mathbf{I}_j, i, j \in V, \\ P_{c5}(i,j) & \text{if } \mathbf{q}' = \mathbf{q} - \mathbf{I}_i + 2\mathbf{I}_j, i, j \in V, i \neq j, \\ P_{c6}(i,j,k) & \text{if } \mathbf{q}' = \mathbf{q} - \mathbf{I}_i + \mathbf{I}_j + \mathbf{I}_k, i, j, k \in V, i \neq j \neq k, \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

$$P\{\mathbf{e}' \mid \mathcal{L}', \mathcal{L}, \mathbf{e}, a\} = \begin{cases} \mathbb{1}\{\mathbf{e}' = \mathbf{e} - \mathbf{I}_i e_{tx} - \mathbf{I}_j e_{rx}\} & \text{if } \Delta\mathcal{L} = (i, j, \varnothing) \in (\mathcal{L} \cup a)^{so}, j \in V^*, \\ \mathbb{1}\left\{\mathbf{e}' = \mathbf{e} - \mathbf{I}_i e_{init}^{CT} - \sum_{k \in \mathcal{H}(i)} \mathbf{I}_k e_{co}^{CT}\right\} & \text{if } \Delta\mathcal{L} = (i, 0, \mathcal{H}(i)) \in (\mathcal{L} \cup a)^{vo}, i \in V, \\ \mathbb{1}\{\mathbf{e}' = \mathbf{e}\} & \text{if } \Delta\mathcal{L} = \varnothing, \\ 0 & \text{otherwise.} \end{cases} \tag{21}$$

$$P\{\mathcal{L}', \mathbf{q}', \mathbf{e}' \mid \mathcal{L}, \mathbf{q}, \mathbf{e}, a\} = P\{\mathbf{q}' \mid \mathcal{L}', \mathbf{e}', \mathcal{L}, \mathbf{q}, \mathbf{e}, a\} P\{\mathcal{L}', \mathbf{e}' \mid \mathcal{L}, \mathbf{q}, \mathbf{e}, a\} \tag{22}$$

$$= P\{\mathbf{q}' \mid \mathcal{L}', \mathbf{e}', \mathcal{L}, \mathbf{q}, \mathbf{e}, a\} P\{\mathbf{e}' \mid \mathcal{L}', \mathcal{L}, \mathbf{q}, \mathbf{e}, a\} P\{\mathcal{L}' \mid \mathcal{L}, \mathbf{q}, \mathbf{e}, a\} \tag{23}$$

$$= P\{\mathbf{q}' \mid \mathcal{L}', \mathcal{L}, \mathbf{q}, a\} P\{\mathbf{e}' \mid \mathcal{L}', \mathcal{L}, \mathbf{e}, a\} P\{\mathcal{L}' \mid \mathcal{L}, a\} \tag{24}$$

Algorithm 1: SSP-MILP Algorithm

input:
$N\theta$ = Minimum Sum energy in states $\mathcal{S} \setminus \mathcal{S}_t$;
Ne^{max} = Maximum Sum energy in states $\mathcal{S} \setminus \mathcal{S}_t$;
$\mathcal{J}(\mathbf{s}) = 0, \forall \mathbf{s} \in \mathcal{S}_t$;
output: $\mathcal{J}(\mathbf{s}), \forall \mathbf{s} \in \mathcal{S} \setminus \mathcal{S}_t$;
begin
  Find transmission link set using Bron-Kerbosch alg. ;
  $m \leftarrow N\theta$;
  while $m \leq Ne^{max}$ **do**
    Solve Bin-ball problem to obtain S_m.
    Solve MILP for Subproblem-1 to obtain $\mathcal{J}(\mathbf{s})$, $\mathbf{s} \in S_m$.
    $m \leftarrow m + 1$.
  end
end

large network is typically reduced into isolated sections of small networks in the time domain, to reduce interference and collisions, i.e., only a small portion of a network is awake during a period of time. Second, the topology captures the essence of the *energy hole* problem.

Fig.3 depicts the optimal lifetime for non-CT and CT networks. We observe the lifetimes are linearly increasing with the battery capacity. We also observe that the performance of CT network is significantly higher than that of the non-CT network. For example, with battery capacity of 10 units, the lifetime improvement factor of CT network is 1.89 with 1 cooperator. The lifetime with larger battery capacities is obtained from linear regression in Fig.4. The lifetime growth rate w.r.t. battery capacity is 1.0 for non-CT, 1.6 for CT with 2 cooperators, and 2.1 for CT with 1 cooperator.

Fig.5 shows the computed lifetime for different packet arrival rates (PARs). The PAR[6] is normalized with the packet length, i.e., the number of new exogenous packets per node during a packet duration. The PARs where the curves start to become flat represent the PAR thresholds, beyond which the numerical results are accurate. Fig.5 demonstrates that our model is accurate for a very large range of arrival rates.

VI. Conclusions

We present and validate a novel MDP framework to model the lifetime of *multihop* wireless sensor networks, for both non-CT and CT networks. The MDP jointly considers MAC layer link constraints, packet transfers in routing layer, and energy evolution dynamics. A new algorithm that exploits the Stochastic Shortest Path structure and the Mixed Integer Linear Programming is proposed to efficiently solve the problem.

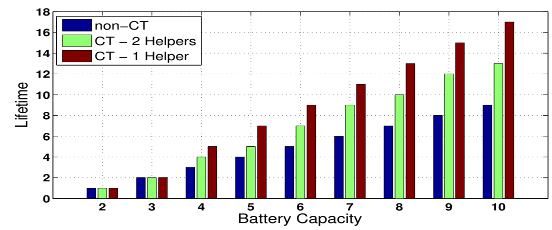

Figure 3: Optimal lifetime for non-CT network and CT network, as battery capacity varies. $N = 4$ (number of nodes). $N_H = 1$ or 2 (required number of cooperators). $e_{th} = 1,\ e_{tx} = e_{rx} = 1,\ e^{CT}_{init} = 1,\ e^{CT}_{co} = 2,\ q^{max} = 1$.

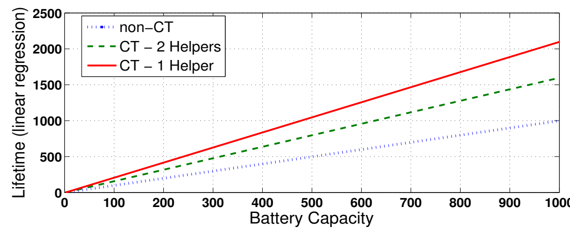

Figure 4: Optimal lifetime for non-CT network and CT network (from linear regression). $N = 4$ (number of nodes). $e_{th} = 1,\ e_{tx} = e_{rx} = 1,\ e^{CT}_{init} = 1,\ e^{CT}_{co} = 2,\ q^{max} = 1$.

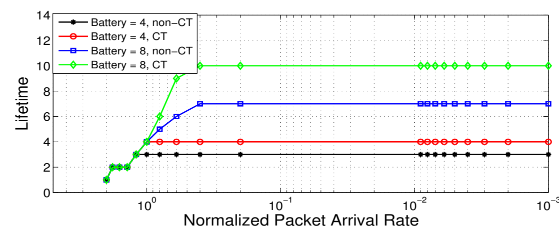

Figure 5: Lifetime for non-CT network and CT network, as the normalized packet arrival rate (PAR) varies. $N = 4,\ N_H = 2$. $e_{th} = 1,\ e_{tx} = e_{rx} = 1,\ e^{CT}_{init} = 1,\ e^{CT}_{co} = 2,\ q^{max} = 1$.

References

[1] J. Li and P. Mohapatra, "An analytical model for the energy hole problem in many-to-one sensor networks," in *Proc. IEEE VTC 2005*.

[2] J. W. Jung and M. A. Ingram, "Residual-Energy-Activated Cooperative Transmission (REACT) to Avoid the Energy Hole," in *Proc. IEEE Int Communications Workshops (ICC) Conf*, 2010, pp. 1–5.

[3] J. Lin and M. A. Ingram, "SCT-MAC: A Scheduling Duty Cycle MAC protocol for Cooperative Wireless Sensor Networks," *in IEEE International Conference on Communications (ICC)*, June 2012.

[4] ——, "OSC-MAC: Duty Cycle Scheduling and Cooperation in Multi-hop Wireless Sensor Networks," *in Proc. IEEE WCNC*, April 2013.

[5] J.-H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," in *IEEE INFOCOM 2000.*, vol. 1, pp. 22–31.

[6] J. W. Jung and M. Ingram, "On the Optimal Lifetime of Cooperative Routing for Multi-hop Wireless Sensor Networks," in *IEEE Wireless Communications and Networking Conference (WCNC)*, April 2013.

[7] Y. Chen, Q. Zhao, V. Krishnamurthy, and D. Djonin, "Transmission Scheduling for Optimizing Sensor Network Lifetime: A Stochastic Shortest Path Approach," *Signal Processing, IEEE Transactions on*, vol. 55, no. 5, pp. 2294 –2309, may 2007.

[8] B. Nardelli and E. Knightly, "Closed-form Throughput Expressions for CSMA Networks with Collisions and Hidden terminals," in *INFOCOM, 2012 Proceedings IEEE*, march 2012, pp. 2309 –2317.

[9] F. Liu, J. Lin, Z. Tao, T. Korakis, E. Erkip, and S. Panwar, "The Hidden Cost of Hidden Terminals," in *Proc. IEEE ICC*, 2010, pp. 1–6.

[10] R. Boorstyn, A. Kershenbaum, B. Maglaris, and V. Sahin, "Throughput Analysis in Multihop CSMA Packet Radio Networks," *Communications, IEEE Transactions on*, vol. 35, no. 3, pp. 267 – 274, mar 1987.

[11] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York, USA: John Wiley & Sons., 1994.

[12] V. Bui, W. Zhu, A. Botta, and A. Pescape, "A Markovian Approach to Multipath Data Transfer in Overlay Networks," *Parallel and Distributed Systems, IEEE Trans. on*, vol. 21, no. 10, pp. 1398 –1411, oct. 2010.

[13] C. Bron and J. Kerbosch, "Algorithm 457: finding all cliques of an undirected graph," *Commun. ACM*, vol. 16, no. 9, pp. 575–577, 1973.

[6] For example, if a node generates a packet every 100 seconds, and the packet transmission time is 100ms, then the PAR is 0.001.