Performance Evaluation of an IEEE 802.15.4 Sensor Network with a Star Topology[†]

Chandramani Kishore Singh[‡] and Anurag Kumar[‡]

***Abstract*— One class of applications envisaged for the IEEE 802.15.4 LR-WPAN (low data rate - wireless personal area network) standard is wireless sensor networks for monitoring and control applications. In this paper we provide an analytical performance model for a network in which the sensors are at the tips of a star topology, and the sensors need to transmit their measurements to the hub node so that certain objectives for packet delay and packet discard are met. We first carry out a saturation throughput analysis of the system; i.e., it is assumed that each sensor has an infinite backlog of packets and the throughput of the system is sought. After a careful analysis of the CSMA/CA MAC that is employed in the standard, and after making a certain decoupling approximation, we identify an embedded Markov renewal process, whose analysis yields a fixed point equation, from whose solution the saturation throughput can be calculated. We validate our model against ns2 simulations (using an IEEE 802.15.4 module developed by J. Zheng [13]). We then show how the saturation analysis can be used to obtain an analytical model for the finite arrival rate case. This finite load model captures very well the qualitative behavior of the system, and also provides a good approximation to the packet discard probability, and the throughput.**

Keywords: wireless sensor networks, performance analysis, LR-WPANs

I. Introduction

Low rate-wireless personal area networks (LR-WPANs) are designed to serve a variety of applications with a focus on enabling wireless sensor networks. The IEEE 802.15.4 standard [1] has evolved to realize the physical (PHY) and multiple access control (MAC) layers of such LR-WPANS. The ZigBee alliance has developed the network and upper layers [2]. The overall objective of our work reported here is to analyse the performance of such networks for industrial sensing and measurement applications. The aim is to replace existing wired sensor networks (based, e.g., on the Fieldbus standard) with wireless ad hoc sensor networks. The end to end applications, however, will initially remain unchanged. Hence the concern is whether the wireless network will be able to carry the measurement and alarm traffic with the same level of performance as the wired network.

In this paper we provide the results of our analysis of a star topology sensor network based on the IEEE 802.15.4 standard. Here we limit our work to the situation in which packets flow only from the sensors to the head of the hub of the star (i.e., the PAN coordinator). We first obtain the saturation throughput of the network. Then we provide some results on performance with finite rate arrivals of measurements. The following is a preview of our main contributions and findings.

1) We provide a fixed point analysis, based on a decoupling approximation as in [8] and [3] for the saturation throughput analysis of IEEE 802.15.4 networks. This analysis captures the saturation throughput with a maximum error of 5% (see Figure 16).
2) We find that the design of the CSMA/CA MAC in IEEE 802.15.4 is such that the aggregate saturation throughput decreases sharply with the number of nodes. We show that, staying within the framework of the standard, it is possible to modify the backoff parameters so that the saturation throughput decreases only slightly when the number of nodes increases. It is also found that packet discard probability is much reduced after these modifications (see Figures 18 and 19).
3) A simple heuristic finite load analysis that is based on our saturation analysis is performed. Simulations show that the analysis captures very well the qualitative behaviour of delay, throughput, and packet discard probabilities, and for the latter two also provides a good analytical approximation (see Figures 21 and 22).

Related Literature: There is little published literature on the analytical modeling of IEEE 802.15.4 networks. Zheng and Lee [9] have provided a qualitative as well as quantitative overview of the standard. Kinney has provided description of ZigBee technology [7], along with a comparative study of ZigBee with Bluetooth. In [14], Zheng and Lee report on a module that they have developed for the ns2 simulator ([13]), and provide results from several sets of simulation experiments. In [5], Golmie et al. have provide a simulation study of the suitability of the IEEE 802.15.4 standard for the medical environment. They have focused on scalability issues and also have evaluated the effect of packet segmentation as well as backoff parameters on performance metrics. In [10], the authors have also done a simulation based performance evaluation of the IEEE 802.15.4 slotted CSMA-CA protocol. They discuss some of the throughput-energy-delay trade offs inherent in this MAC protocol. Timmons and Scanlon have done the very first analytical modeling for the IEEE 802.15.4 single cell network in relation to medical sensor body area networks. Their much simplified analysis [12] focuses on long-term power consumption of devices.

Our analytial approach is in the spirit of that performed in [8] (see also [3]) for a single cell saturated IEEE 802.11 WLAN. This saturation analysis is used to develop a finite load analysis using an approach suggested in [6, Chapter 4].

[†]The work was sponsored by the Honeywell Technology Solutions Lab.

[‡]Department of Electrical Communication Engg., Indian Institute of Science, Bangalore, 560 012, INDIA; email: cksingh, anurag@ece.iisc.ernet.in

II. LR-WPAN Architecture and Specifications

A. An Overview

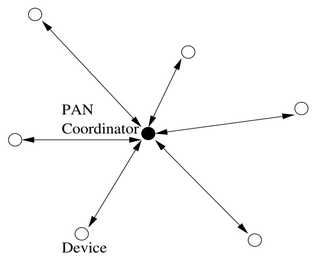

Fig. 1. A star topology LR-WPAN sensor network

Figure 1 shows a star topology sensor network comprising a PAN coordinator and several sensors as leaf nodes. Since we consider only a simple star topology, with flow of traffic only from the leaf nodes to the hub, we need to consider only the PHY and MAC standards. Throughout we assume that we are working in the 2.45GHz band and hence the PHY data rate is 250 Kbps, the symbol rate is 62.5 symbols/second; hence the symbol time is $16\mu s$. In practice wireless transceivers are always half duplex. Hence the IEEE 802.15.4 devices require a finite amount of time to switch between transmission and reception. This time is denoted by *aTurnaroundTime* in the standard and is equal to 12 symbol times.

We now turn to the MAC specifications. The PAN coordinator can optionally work with time slots defined through a superframe structure (see Figure 2). This option permits a synchronous operation of the network so that nodes can go to sleep and wake up at designated times. We assume this in our work. Each superframe starts with the transmission of a beacon and has active and inactive portions. The active portion is composed of three parts: a beacon, a contention access period (CAP), and a contention free period (CFP). A CAP commences immediately after the beacon. All frames, except acknowledgment frames and any data frame that immediately follows the acknowledgment of a data request command, transmitted in the CAP, must use a slotted CSMA-CA mechanism to access the channel.

Transmitted frames are always followed by an IFS period. Frames(MPDUs) of length up to *aMaxSIFSFrameSize* are followed by a SIFS period of duration of at least *aMinSIFSPeriod* symbols, otherwise a LIFS of a duration at least *aMinLIFSPeriod* symbols follows (see Figure 3).

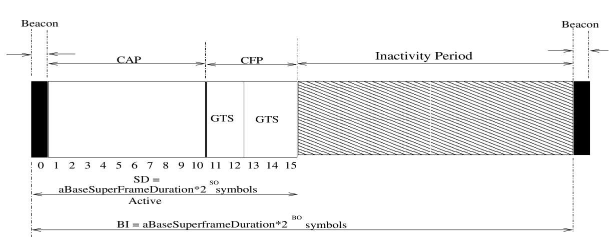

Fig. 2. The IEEE 802.15.4 superframe structure.

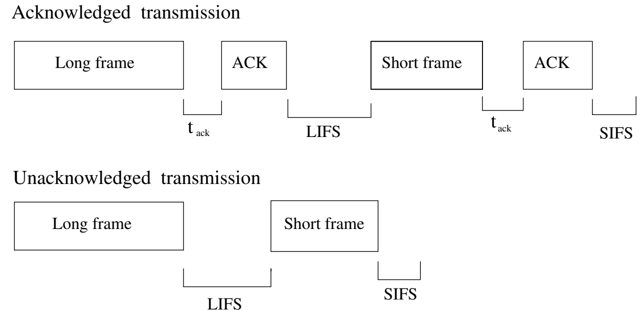

Fig. 3. Illustration of LIFS and SIFS

B. The Slotted CSMA-CA Algorithm

The CSMA-CA algorithm is implemented using units of time, called backoff periods, each of length *aUnitBackoffPeriod* (= 20 symbol times = 0.32 ms). Note that 10 bytes can be transmitted in one backoff period. In slotted CSMA-CA, the backoff period boundaries of every device in the PAN are aligned with the superframe slot boundaries of the PAN coordinator, and transmissions begin on the boundary of a backoff period.

Each device maintains three variables for each transmission attempt: NB, CW and BE. NB is initialised to 0, and counts the number of additional backoffs the algorithm has to do while attempting the current transmission. CW is the number of backoff periods, that need to be clear of channel activity before the transmission can commence. MAC ensures this by performing *clear channel assessment* (CCA) at the boundary of CW consecutive backoff periods[1]. CW is set to 2 before each transmission attempt. BE is the backoff exponent. Before performing the CCAs a node a takes backoff of $random(0, 2^{BE} - 1)$ backoff periods. In slotted systems with *macBattLifeExt* set to TRUE, BE is initialized to lesser of 2 and the value of *macMinBE*. In other cases it is initialized to *macMinBE*. If either of the CCAs fails, CW is reset to two and both NB and BE are increased by one, ensuring that BE does not exceed *aMaxBE*. If the value of NB is greater than *macMaxCSMABackoff*, the CSMA-CA algorithm terminates with a *Channel Access Failure* status. The concerned packet is discarded after a *Channel Access Failure*. Default values of these parameters as in standard are given in following table.

If both the CCAs from a node succeed, it will transmit the packet. This may result either in a successful transmission or a collision. A A successful transmission is always accompanied by the reception of a MAC acknowledgment. A MAC acknowledgment is of fixed length, 11 bytes. A transmitting node always waits for acknowledgment for *macAckWaitDuration* before declaring a collision. If a packet collides with some other packet while being transmitted, it is retransmitted with all backoff parameters set to their initial values. A packet is

[1]Energy saving is an important consideration in sensor networks. In the 802.11 standard nodes keep their receivers on even during backoff periods so that they can sense any transmission and freeze their counters during activity periods. However, carrier sensing also requires energy. In the 802.15.4 standard, during backoff, a node's receiver can shut down and CCA is performed only after backoff is finished, thus saving energy.

retransmitted at most *aMaxFrameRetries* times after transmission failures due to collisions, before being discarded.

The default values of these parameters, as in the standard, are given in the following table.

Parameter	Value
aMaxBE	5
aMaxframeRetries	3
macMaxCSMABackoffs	4
macMinBE	3

III. Modeling Simplifications and Observations

In Sections IV and V we develop an analytical model for calculating the saturation throughput of an IEEE 802.15.4 star network. We analyse a star network with n sensor devices. The PAN coordinator acts as a data sink. The network is assumed to be beacon enabled. All sensor nodes contend to send data to the PAN coordinator. By saturation throughput it is meant that all the nodes always have packets to send, and hence always contend for access. Saturation throughput is one measure of system capacity, and we will see how it can be used to develop an approximate analysis for finite arrival rates.

A. Modeling Simplifications

Since we are interested in an application in which measurements continuously flow from the sensors to the PAN coordinator, it is assumed that the active part of the superframe is equal to the beacon interval. Also no CFPs are assumed. Further, we ignore the time taken to transmit the beacon and all the time wasted at the end of each beacon interval due to nodes not being able to complete their transmissions in the fragment of time left at the end of a beacon interval. Thus the channel time is assumed to comprise of an uninterrupted sequence of backoff intervals.

Since we are analyzing the star topology, which has only one hop transmissions, the ZigBee routing algorithm does not come into the picture. We assume direct communication between the nodes and the PAN coordinator. It is also assumed that none of the devices disassociates during the whole traffic flow, and also that communication failures never cause a device to conclude that it has been orphaned.

The CSMA-CA algorithm described here assumes battery life extension subfield set to 0, so that backoff countdown can occur throughout the active portion of the superframe and the frame transmission also can start at any of the backoff period boundaries throughout the active portion of the superframe.

We assume also that all nodes are perfectly synchronized at backoff period boundaries.

B. Some Observations

Whenever a node has a packet to transmit, it starts a random backoff. When a node completes its backoff, it seeks a reservation of the channel. For that purpose, following the backoff, it performs a CCA, at the start of the next backoff period, to see whether the channel is free. A CCA lasts for 8 symbol times and if the channel is found to be free at the end of the 8^{th} symbol time, CCA succeeds. This is because if the channel becomes free even at the end of the 8^{th} symbol time, the node can turn its transmitter on within the remaining 12 symbol times and can start a new activity at the start boundary of the next backoff period. This implies that if an activity (successful transmission or collision) finishes before the end of the 8^{th} symbol time in a backoff period, the channel appears to be *virtually free* from the point of view of all the nodes not involved in the activity.

In case the first CCA succeeds, the node waits until the start of the next backoff slot and performs one more CCA. If the channel is again found to be free, the node starts transmission at the start of the next backoff period boundary. In case any of the two CCAs fails, the node again enters the backoff state and repeats the steps mentioned above, at the end of the backoff interval. For the first few attempts for a packet, the mean backoff interval increases multiplicatively with each channel access, the rate of increase being governed by the backoff multiplier (i.e., 2), but later the backoff interval remains fixed until the current attempt cycle terminates. The reason for using two CCAs becomes clear from the remaining discussion in this subsection.

A successful transmission is always accompanied by reception of a MAC acknowledgment of length 11 bytes (see Figure 4, where the last byte of the ACK is shown spilling over into the second backoff period). Once a node finishes reception of data, it needs a time t_{ack} (see Figure 3) before its transmitter is turned on, and then it starts transmission of the MAC ACK. Since, transmission of an ACK can start at a backoff period boundary only, t_{ack} can have values in the range: *aTurnaroundTime* $\leq t_{ack} \leq$ *aTurnaroundTime* + *aUnitBackoffPeriod*. The whole transaction also includes an interframe space time after transmission of the MAC acknowledgment (see Figure 3).

In case data transmission finishes before the end of the 8^{th} symbol time in its last backoff period, the receiver can turn its transmitter on (during the remaining 12 symbol times) before the start of the next backoff period and, hence, it can start the transmission of the ACK at the next backoff period boundary. But if data requires more than 8 symbol times in its last backoff period, the receiver has less then 12 symbol times in that backoff period and hence the turn around time spills over into the next backoff period. So, the ACK has to wait one more entire backoff period. In either case the channel becomes virtually free for exactly one backoff period. Denote that backoff period by t_{ack^*}. Figure 4 shows these situations.

There will be a collision only if two or more nodes start their first CCA (in the sequence of two CCAs) at the same backoff boundary. A transmitting node always waits for acknowledgment for *macAckWaitDuration* before declaring a collision. This is the worst case delay which can occur in reception of acknowledgment. Denote the ACK transmission time by $\mathcal{T}_{ack}$. Then *macAckWaitDuration* = $\max(t_{ack})+\mathcal{T}_{ack}$ = *aTurnaroundTime* + *aUnitBackoffPeriod* + $\mathcal{T}_{ack}$. This situation is depicted in Figure 5, where nodes 1 and 2 are shown to collide.

While a node performs CCAs, if some activity is going on

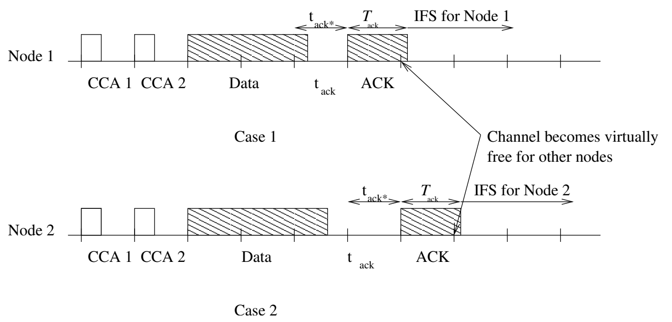

Fig. 4. A successful transmission.If the amount of time occupied by data in its last backoff period is less than 8 symbol times then 12 symbol times remain in the backoff period, sufficient to turn around and send an ACK; then Case 1 occurs. Otherwise the turn around time spills over into the next backoff period and the ACK must start at the beginning of the next backoff period boundary; hence, Case 2 occurs.

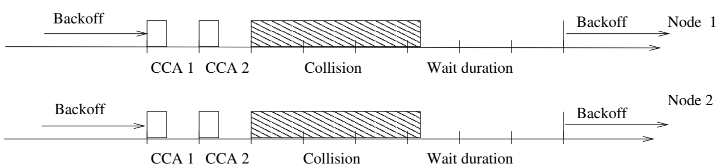

Fig. 5. A collision between two nodes. Both complete their backoffs at the same backoff period boundary, and both CCAs of both nodes find the channel idle. Then the packet transmissions collide.

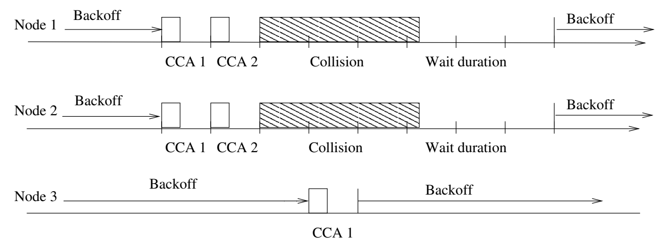

Fig. 7. Node 3 attempts while channel is in collision and not free even virtually. Its first CCA fails and it enters its next backoff cycle for the same packet.

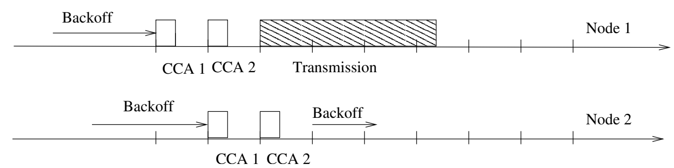

Fig. 8. Node 2 attempts while Node 1 is in a second CCA. Its second CCA fails even when the first has succeeded. Node 1's transmission carries on, while Node 2 enters into the next backoff

and the channel is not virtually free, its first CCA itself will fail. This possibility is shown through Figures 6 and 7.

On the other hand, when a node starts a CCA, if the channel is either in second CCA (CCA 2) or in t_{ack^*} due to some node, then the first CCA (CCA 1) from the node will succeed, but this node's CCA 2 will fail. Figures 8 and 9 show the situations where the first CCA succeeds while the second fails for Node 2.

C. Observations about the IEEE 802.15.4 ns2 Module

A detailed look at IEEE 802.15.4 ns2 module [13] revealed that there are a few inconsistencies between the module and the IEEE 802.15.4 standard [1]. We made changes in the module to remove the following discrepancies.

1) A node performing CCA decides whether the CCA has succeeded only at the end of the 8^{th} symbol time in a backoff period. Once the second CCA succeeds, the remaining 12 *symbol times* (which is equal to *aTurnaroundTime*) left in that backoff period are sufficient for the node to turn its transmitter on. In the ns2 module, a node spends one extra backoff period before starting actual transmission, whenever its second CCA succeeds.
2) Whenever a node's frame collides, the attempt counter is increased by one until the counter exceeds *aMaxFrameRetries*, after which the frame is discarded. Unlike the standard, in the ns2 module a node resets this counter when the attempt process of a packet spills over into a new superframe.
3) In case a backoff failure occurs while attempting for a packet, the packet is discarded. But in the ns2 module a packet is reattempted indefinitely often after backoff failures, and is discarded only if it faces more than *aMaxFrameRetries* collisions.

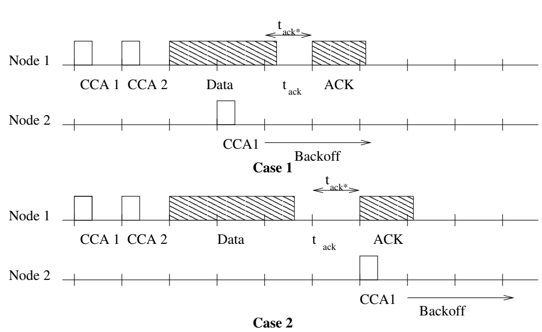

Fig. 6. Node 2 attempts while data or an acknowledgment are being transmitted in Case 1 and Case 2 respectively. In both the cases the channel is not virtually free. Node 2's first CCA fails and it enters its next backoff cycle for the same packet.

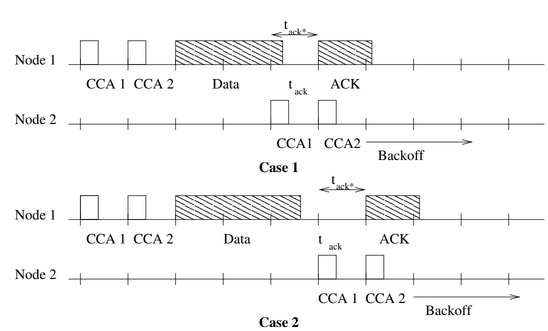

Fig. 9. Node 2 attempts while channel is in t_{ack^*} . Its second CCA fails even when the first has succeeded. Node 1's transmission carries on, while Node 2 enters into the next backoff

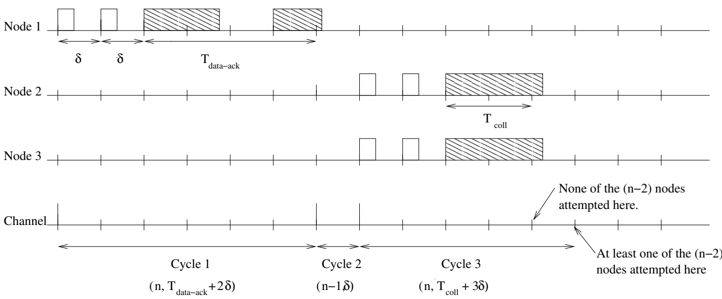

Fig. 10. A snapshot of evolution of the activity at three nodes. The fourth time-line superposes these activities, thus depicting a cyclical evolution for the aggregate system.

IV. The Stochastic Model and A Fixed Point Equation

A. A Cyclic Evolution

Time is divided into contiguous backoff periods whose duration is denoted by δ. All node activities are initiated at backoff period boundaries. We need to study each node's individual behavior and also the aggregate channel activity. From the point of view of the channel activity, we can define certain cycles (see the last time line in Figure 10). There could be a succession of idle backoff periods. An idle channel period ends when both of the successive CCAs of one or more nodes are successful. Once this happens, the evolution of the channel activity in the cycle becomes deterministic; i.e., subsequently there may be a successful transmission (as in case of Node 1 in Figure 10), or a collision between two or more nodes (like Nodes 2 and 3 in Figure 10).

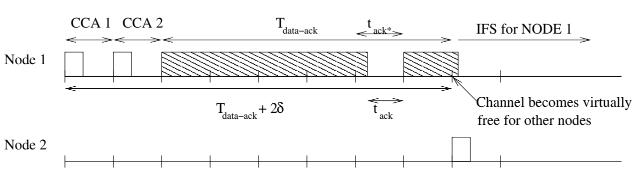

Fig. 11. A successful transmission. It lasts for $T_{data-ack} + 2\delta$ duration, the channel is virtually free at the time shown and only $n-1$ nodes are available for attempt in the following backoff period.

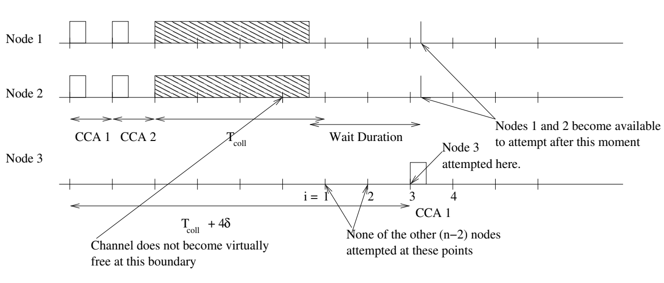

Fig. 12. A cycle containing collisions. Its length is $T_{coll} + 4\delta$ because Node 3 has attempted at the 3^{rd} backoff period boundary after collision is over. Only $n-2$ nodes were available there to attempt at that point for new cycle.

Suppose the network consist of n contending nodes, excluding the PAN coordinator. The length of a cycle depends on *the number of nodes available to attempt* in the first backoff period of the cycle. A cycle contains a single idle backoff period of length δ, if none of the available nodes attempts to sense the channel at the start boundary of this backoff period (see Cycle 2 in Figure 10). It leaves all those nodes free to attempt in the next cycle.

In Figure 11 we show a successful transmission cycle. Let us examine this carefully. The total busy period of the transaction is shown as $T_{data-ack}$. Acknowledgments always start at the boundary of a backoff period and being of fixed length 11 bytes (i.e., 22 symbols) consume only 2 symbol times in their last backoff period. Thus, at the start boundary of this backoff period, the channel becomes *virtually free* for other nodes because a CCA that starts at this backoff period boundary (being of 8 symbol times duration) will find the channel clear, and hence a new cycle might be started by other nodes in this backoff period. Thus, $T_{data-ack}$ does not include this backoff period. Define $T_{data-ack^*}$ as the portion of $T_{data-ack}$, excluding t_{ack^*} (recalling t_{ack^*} from Section III-B) i.e.,

$$T_{data-ack^*} = T_{data-ack} - t_{ack^*}$$

It is seen that, a cycle of successful transmission lasts for $T_{data-ack} + 2\delta$ duration and, only $n-1$ nodes are available for attempt in the following backoff period. Note from Figure 10 that the channel is viewed as being in an idle cycle (Cycle 2) even though the transmission of Node 1 is not complete.

In Figure 12 we depict a collision cycle. If there are 2 or more nodes available to attempt at the beginning of a cycle, there is a possibility of collision. If there is a collision of k nodes, it continues for T_{coll} duration. During its last backoff period, if a collision consumes less than or equal to 8 symbol times, again the channel remains virtually free from the point of view of nodes not involved in collision; then, this backoff period is not included in T_{coll}. After the collision, all the k senders wait for acknowledgment for $macAckWaitDuration$ before declaring *channel access failure*. Define a positive integer J such that

$$J = \frac{\lfloor \mathcal{T}_{coll} + macAckWaitDuration \rfloor}{\delta} + 2 - \frac{T_{coll}}{\delta}$$

where $\mathcal{T}_{coll}$ is the actual time spent in collision by a node. A careful look at various parameters reveals that the only possible values of J are 4 and 5. Thus a cycle containing a collision activity has one of the following three possibilities.

- Case 1: $k < n$, and one or more of the $n-k$ nodes not involved in the collision perform successful CCAs, while k nodes involved in collision are still waiting for acknowledgments. In this case the length of the current cycle will be $T_{coll} + j\delta, j \in \{2, 3, ..J\}$ with, $n-k$ nodes being available to contend for next cycle. Also the next cycle cannot be an idle cycle.
- Case 2: $k < n$, and all the k nodes involved in the collision finish with their *macAckWaitDuration*s and none of the other $n-k$ nodes attempts for a CCA in this duration. In this case the length of the current cycle will be $T_{coll} + (J+1)\delta$, and all the n nodes will be available to contend for next cycle.
- Case 3: $k = n$. There will not be any node available to attempt after the collision is over. The current cycle will

last for $T_{coll} + (J+1)\delta$ duration, and all the n nodes will be available to contend for next cycle.

These situations are illustrated by Figure 12, which corresponds to $J = 4$. Here the current cycle may be of lengths $T_{coll} + (i+1)\delta, i = 1, 2, 3,$ provided that some of the nodes other than 1 or 2, attempt at the i^{th} backoff period boundary after the collision is over. The cycle length will be $T_{coll} + 5\delta$ if none of other nodes attempt while nodes 1 and 2 are waiting for acknowledgments. The figure shows the case where the cycle length is $T_{coll} + 4\delta$. In Figure 10 the channel enters into Cycle 4, while Nodes 2 and 3 are still waiting for acknowledgments.

B. The Stochastic Model

As shown in Section IV-A, the cycles defined are always multiples of the backoff period. Denote the backoff period boundaries by $t_k = k\delta, k \geq 0$. Then denote the start times of the cycles by the random times $T_i, i \geq 0$, with $T_i \in \{t_j : j \geq 0\}$, and $T_0 (= 0) < T_1 < T_2 < \cdots$. Associated with each $T_i, i \geq 0$, is a random variable $X_i \in \{1, 2, ...n\}$, which is the number of nodes available to attempt at the instant T_i. The cycles are indexed by $i \geq 1$, with cycle i being the interval $[T_{i-1}, T_i)$. Denote the cycle length by

$$U_i = T_i - T_{i-1}$$

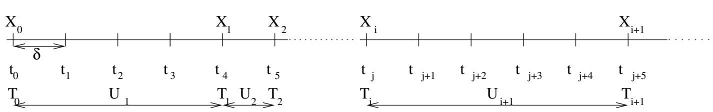

Fig. 13. Channel cycles and notation for the Markov renewal process

We draw the following conclusions from the discussion of Section IV-A.

- If $X_i = n$, then Cycle i may comprise a successful transmission, a collision, or the cycle may be an idle one, depending upon the number of nodes that attempt.
- If $X_i = n - 1$ (as would happen after a success cycle), then cycle i may have a successful transmission, a collision, or the cycle may be an idle one, depending upon the number of nodes that attempt.
- If $X_i < n-1$, it means we are in a case like that shown in Figure 12, i.e., at least one of these nodes has attempted, and the following cycle cannot be an idle one. It can have a successful transmission or a collision depending upon how many nodes have attempted.

It is seen that, if the number of nodes available to attempt at the beginning of the cycle is known, the evolution of the cycles in the future does not depend on the past, i.e., the random vector (U_{i+1}, X_{i+1}) and the random vector $((X_0, T_0), (X_1, T_1), \cdots\cdots\cdots (X_{i-1}, T_{i-1}), T_i)$ are independent, given X_i. Hence, although the cycles are not independent, $(X_i, T_i), i = 0, 1, 2......$ is a Markov renewal process. To analyse this we need the transition probabilities

$$P(U_{i+1} = u, X_{i+1} = k | X_i = k')$$

for all possible values of u, k and k'. Also, $\{X_i, i \geq 0\}$ will be a Markov chain. We can obtain the transition probability matrix $\mathbf{M}$ for this Markov chain, and hence we can compute the steady state probabilities $\pi_k, 1 \leq k \leq n$.

Given the number of nodes available to attempt at the beginning of a cycle, the conditional expectation of the corresponding cycle length can be developed, and we can define the following quantities. For $1 \leq k \leq n$ (and any i) define,

$$E_k U = \sum_u u \sum_{k'} P(U_{i+1} = u, X_{i+1} = k' | X_i = k)$$

Then, the expected duration of a cycle will be given as

$$EU = \sum_{k=1}^{n} \pi_k E_k U$$

We can also determine the conditional expected durations for which channel is either in second CCA, $T_{data-ack}$ or T_{coll}, in a cycle. More generally, suppose, in the i^{th} cycle the channel remains in event e for an amount of time for $R_i^{(e)}$. Thus, $R_i^{(e)}$ can be considered as a "reward" (corresponding to the occurrence of event e) in the i^{th} cycle. It can also be shown that, for each event e of interest, $R_i^{(e)}$ will be a *function* of (U_i, X_i) (see Table I). By this we mean that for each event of interest e, there is a function $r^{(e)}(u, k')$, such that, for all possible values of u, k' and any i, if $(U_i, X_i) = (u, k')$ then $R_i^{(e)} = r^{(e)}(u, k')$. Note that $\{R_i^{(e)}, i \geq 1\}$ are such that, given X_i, $R_{i+1}^{(e)}$ and $((X_0, T_0), (X_1, T_1), \cdots, (X_{i-1}, T_{i-1}), T_i)$ are independent. Then we define (for any i)

$$E_k R^{(e)} = \sum_u \sum_{k'} r^{(e)}(u, k') P(U_{i+1} = u, X_{i+1} = k' | X_i = k)$$

Now, the expected duration for which the channel remains in event e in a cycle, can be obtained as

$$E(R^{(e)}) = \sum_{k=1}^{n} \pi_k E_k R^{(e)}$$

Let $R^{(e)}(t)$ be the duration during $[0, t]$ for which the channel is in event e. Then by a regenerative argument (or the Markov renewal reward theorem) we see that, with probability 1,

$$\lim_{t \to \infty} \frac{R^{(e)}(t)}{t} = \frac{\sum_{k=1}^{n} \pi_k E_k R^{(e)}}{\sum_{k=1}^{n} \pi_k E_k U}$$

Hence various event rates in the system can be determined (e.g., the throughput of good packets can be obtained this way).

C. A Decoupling Approximation

Motivated by the approach in [3] and [8], we propose a decoupling approximation in order to analyse the above process. Each node alternates between periods when it performs backoffs and unsuccessful CCAs and periods when it transmits (successfully or unsuccessfully). Let β denote the rate at which a node's backoffs complete during the time when it is performing backoffs. Unlike the IEEE 802.11 DCF mechanism, here nodes do not freeze their backoff timers when the channel is reserved or when there is a collision. So it is

$r^{(e)}(u,k)$		e					
		$CCA2$	$T_{data-ack}$	T_{coll}	$T_{data-ack^*}$	t_{ack^*}	successfully sent data
(u,k)	(δ, n)	0	0	0	0	0	0
	$(T_{data-ack}+2\delta, n-1)$	δ	$T_{data-ack}$	0	$T_{data-ack}-\delta$	δ	L_{data}
	$(T_{coll}+j\delta, k)$ $j \in \{2,3,\ldots J\}$ $k \in \{1,2,\ldots(n-2)\}$	δ	0	T_{coll}	0	0	0
	$(T_{coll}+(J+1)\delta, n)$	δ	0	T_{coll}	0	0	0

TABLE I

VALUES OF $r^{(e)}(u,k)$ FOR SEVERAL eS AND ALL POSSIBLE (u,k)S. IT CAN BE SEEN THAT $R_i^{(e)}$ CAN BE UNIQUELY DETERMINED, GIVEN (U_i, X_i).

not possible to work with "conditional time" as is done in [8] to facilitate the fixed point analysis for 802.11 WLANs. Instead we proceed as follows. As shown in Figure 10, the channel evolves over cycles. At the end of each cycle we need to determine the activity in the next cycle. The nodes that can attempt in the next cycle are in their backoff periods. We assume that each such node attempts independently in a slot with probability β. Thus, if k nodes can potentially attempt at a backoff period boundary, then we assume that the number of attempts is binomially distributed with parameters k and β. With this assumption, the transitions probabilities of the Markov renewal process, and the conditional expectations defined in Section IV-B can be written down; due to lack of space we do not provide these expressions here (these details are in the full version of the paper available at http://www.ece.iisc.ernet.in/˜anurag). This approach permits us to obtain channel event rates in terms of the unknown value β. Channel event rates, such as the probability of the channel carrying radio activity, are then used to obtain β for a node. This yields a fixed point equation that is developed in Section IV-D. The fixed point equation yields β, which can then be used to obtain the aggregate saturation throughput of the network.

D. A Fixed Point Equation for β

Let us tag a node and obtain its β. A CCA from the tagged node will fail if it finds the channel either in second CCA, $T_{data-ack}$ (see Figure 11) or T_{coll} (see Figure 12). Let α_{CCA2}, $\alpha_{data-ack}$ and α_{coll} be the probabilities of the channel being in second CCA, $T_{data-ack}$ or T_{coll} respectively. Then considering each of these as an event e, we can use the analysis in Section IV-B to obtain their time rates. These durations can be considered as "rewards" associated with various cycles of channel activity. Then, using the analysis in Section IV-B, and noting that once we have tagged a node we need to find the above probabilities for the other $n-1$ nodes, the desired probabilities are given as:

$$\alpha_{CCA2} = \frac{\sum_{k=1}^{n-1}\pi_k E_k R^{(CCA2)}}{\sum_{k=1}^{n-1}\pi_k E_k U} =: H_{CCA2}(n-1,\beta)$$

$$\alpha_{data-ack} = \frac{\sum_{k=1}^{n-1}\pi_k E_k R^{(data-ack)}}{\sum_{k=1}^{n-1}\pi_k E_k U} =: H_{data-ack}(n-1,\beta)$$

$$\alpha_{coll} = \frac{\sum_{k=1}^{n-1}\pi_k E_k R^{(coll)}}{\sum_{k=1}^{n-1}\pi_k E_k U} =: H_{coll}(n-1,\beta)$$

Note that the right hand sides of the above three equations depend on β. Hence α, the probability that a tagged node's CCA will fail, can be given in terms of β as

$$\alpha = H(n-1,\beta)$$

where $H(n-1,\beta) := \alpha_{CCA2} + \alpha_{data-ack} + \alpha_{coll}$, with each term being given as above. Also let $\alpha_{data-ack^*}$ and $\alpha_{t_{ack^*}}$ be the fractions of time, the channel is in $T_{data-ack^*}$ and t_{ack^*} respectively. These quantities can also be calculated as functions of β, in the way shown above. Let

$$\alpha_{data-ack^*} = \frac{\sum_{k=1}^{n-1}\pi_k E_k R^{(data-ack^*)}}{\sum_{k=1}^{n-1}\pi_k E_k U} := H_{data-ack^*}(n-1,\beta)$$

$$\alpha_{t_{ack^*}} = \frac{\sum_{k=1}^{n-1}\pi_k E_k R^{(t_{ack^*})}}{\sum_{k=1}^{n-1}\pi_k E_k U} := H_{t_{ack^*}}(n-1,\beta)$$

Evidently, $\alpha_{data-ack} = \alpha_{data-ack^*} + \alpha_{t_{ack^*}}$.

Having obtained these channel probabilities in terms of β, we now turn to obtaining β in terms of the channel probabilities, thus leading to a fixed point equation. Recall that all the nodes have saturated queues. A node goes through the backoff procedure for its head-of-the-line (HOL) packet. Each such backoff procedure can end in one of three ways: successful transmission, collision, or backoff failure (i.e., none of the CCAs during the backoff procedure succeeds). If there is a success, a fresh backoff procedure is begun for the next packet. If there is a collision, a fresh backoff procedure is begun for the same packet and the packet is discarded after N collisions ($N = 4$ in the standard, as 3 retries are allowed). Upon the discard of a packet a fresh backoff procedure is begun for the next packet. Finally, if there is a backoff failure for a packet, the packet is discarded, and the next packet moves to the HOL position. We seek β, the rate at which backoffs complete during times when the tagged node is performing backoffs, given the various probabilities for the the other $n-1$ nodes, as obtained above. Let us denote

- $K =$ the number of times channel sensing is reattempted in a backoff cycle
- $b_k =$ the mean backoff duration (in backoff periods) before the $(k+1)^{th}$ channel sensing attempt for a packet, $0 \le k \le K$

It must be noted that during a backoff procedure, each backoff will be followed by one or possibly two CCA durations before the next backoff starts following a CCA failure.

Suppose, $A_j^{(k)}$ denotes the number of attempts for the j^{th} backoff procedure after the k^{th} *channel access failure*. Thus,

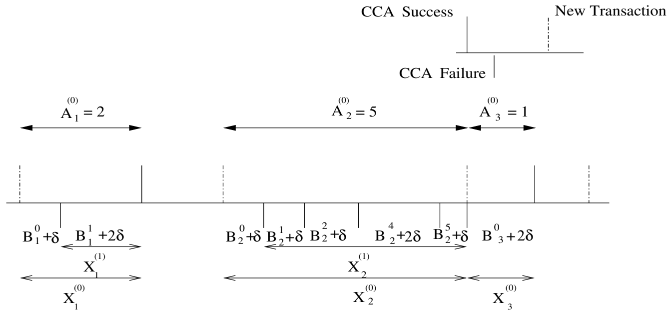

Fig. 14. Evolution of backoff, channel sensing and busy periods for a tagged node. The first backoff procedure had two CCAs, the second of which succeeded. The procedure took an amount of time $X_1^{(0)}$. The second backoff procedure had 5 attempts and ended in backoff failure. The third had an attempt and ended in a successful CCA. Then the backoff rate during backoff times, over this fragment of the node's evolution is $\frac{2+5+1}{X_1^{(0)}+X_2^{(0)}+X_3^{(0)}}$.

$A_j^{(0)}$ will give the total number of attempts for the backoff procedure and the sequence of backoff times will be $B_j^{(k)} (0 \leq k \leq A_j^{(0)} - 1)$, with $E(B_j^{(k)}) = b_k$. Also let $X_j^{(k)}$ be the time duration for j^{th} backoff procedure between the k^{th} *channel access failure* and the end of the backoff procedure. Then $X_j^{(0)}$ will denote the total time occupied by the j^{th} backoff procedure. Evolution of the backoff, channel sensing and busy periods for a node are shown in Figure 14.

Under the decoupling approximation we observe that the sequence $X_j^{(0)}, j \geq 1$, are renewal life times. Viewing the total number of attempts in the j^{th} backoff procedure, $A_j^{(0)}$, as a "reward" associated with the renewal cycle of length $X_j^{(0)}$, we see from the renewal reward theorem that the attempt rate β of a node during its backoff period is given by $\frac{E(A_j^{(0)})}{E(X_j^{(0)})}$. We now derive expression for this ratio in terms of the channel probabilities obtained earlier.

We assume that the CCAs of the tagged node "see" the remaining $n-1$ nodes in steady state and hence the probability that the tagged node's CCA fails is α. Thus

$$E(A_j^{(0)}) = 1 + \alpha E(A_j^{(1)})$$

and, further

$$E(A_j^{(1)}) = 1 + \alpha E(A_j^{(2)})$$

Hence recursing (using $E(A_j^{(K)}) = 1$) we get

$$E(A_j^{(0)}) = \sum_{k=0}^{K} \alpha^k$$

We now turn to $E(X_j^{(0)})$. A node's first CCA will fail if it finds the channel in $T_{data-ack^*}$ or T_{coll}. Its first CCA will succeed while the second will fail if the channel is in the second CCA or t_{ack^*} . Hence,

$$\begin{aligned} E(X_j^{(0)}) &= b_0 + (\alpha_{data-ack^*} + \alpha_{coll})(\delta + E(X_j^{(1)})) \\ &+ (\alpha_{CCA2} + \alpha_{t_{ack^*}})(2\delta + E(X_j^{(1)})) + (1-\alpha)2\delta \end{aligned}$$

Using the fact that

$$\alpha = \alpha_{data-ack^*} + \alpha_{t_{ack^*}} + \alpha_{coll} + \alpha_{CCA2}$$

we get

$$E(X_j^{(0)}) = b_0 + \alpha E(X_j^{(1)}) + (2 - (\alpha_{data-ack^*} + \alpha_{coll}))\delta$$

Similarly

$$E(X_j^{(1)}) = b_1 + \alpha E(X_j^{(2)}) + (2 - (\alpha_{data-ack^*} + \alpha_{coll}))\delta$$

Again recursing, with $E(X_j^{(K)}) = b_K + (\alpha_{data-ack^*} + \alpha_{coll})\delta$ $+ (\alpha_{CCA2} + \alpha_{t_{ack^*}})2\delta + (1-\alpha)2\delta$, we get

$$E(X_j^{(0)}) = \sum_{k=0}^{K} \alpha^k (b_k + (2 - (\alpha_{data-ack^*} + \alpha_{coll}))\delta)$$

Thus, the attempt rate β can be obtained as

$$\begin{aligned} \beta &= \frac{\sum_{k=0}^{K} \alpha^k}{\sum_{k=0}^{K} \alpha^k (b_k + (2 - (\alpha_{data-ack^*} + \alpha_{coll}))\delta)} \\ &:= G(\alpha, \alpha_{data-ack^*}, \alpha_{coll}) \end{aligned}$$

Now, it can be expected that the equilibrium behavior of the system will be characterized by the solutions of the following fixed point equation:

$$\begin{aligned} \beta &= G(H(n-1,\beta), H_{data-ack^*}(n-1,\beta), H_{coll}(n-1,\beta)) \\ &=: \Gamma(\beta) \end{aligned}$$

Since $G(\cdot), H(\cdot), H_{data-ack^*}(\cdot)$ and $H_{coll}(\cdot)$ are continuous functions so is $\Gamma(\cdot)$. Thus $\Gamma(\cdot)$ is a continuous map from $[0,1]$ to $[0,1]$ and hence by Brouwer's fixed point theorem there is a fixed point.

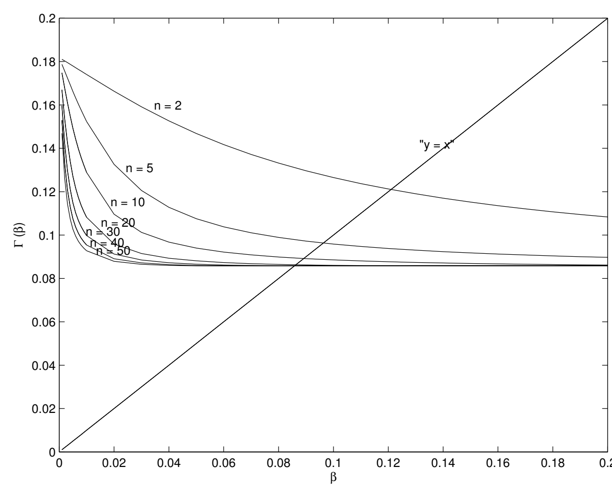

Fig. 15. $\Gamma(\beta)$ vs β for several values of n. The intersections with the "$y = x$" line yield the fixed points.

Numerical solution of the fixed point equation: We assume operation in the 2.45GHz band and hence the PHY data rate is 250 Kbps. The backoff multiplier is $p = 2$ as in the IEEE 802.15.4 standard. In the plots we use the following values: $K = 4$, *macMinBE* $= 3$, *aMaxBE* $= 5$, and $b_0 = 3.5$ backoff periods. The packet size (MSDU) is assumed to be 30 bytes throughout. Figure 15 shows plots of $G(H(n-1,\beta), H_{data-ack^*}(n-1,\beta), H_{coll}(n-1,\beta))$ vs. β. The intersections of the plots with the "$y = x$" line yield the fixed points. We obtain the fixed points by using the **fzero()**

function in MATLAB. At this point in our work we are unable to analytically show uniqueness. However, in all the cases we examined the fixed point was unique. We see that for this set of parameters the attempt rate of the individual nodes remains almost constant (at about 0.086) once the number of nodes exceeds 10.

V. Calculation of Performance Measures

A. Throughput Calculation

To calculate the throughput, we again use the Markov renewal process formulated earlier in Section IV-B. Successfully sent data in a cycle can be considered as yet another "reward" associated with that cycle.

A successful data transmission will take place in cycle i only if $(U_i, X_i) = (T_{data-ack} + 2\delta, n-1)$. Consider L_{data} as the size of a packet. Then the expected amount of data sent in a cycle, having k nodes to attempt at its beginning, is given by:

$$E_k L = L_{data} P(U_{i+1} = T_{data-ack} + 2\delta, X_{i+1} = n-1 | X_i = k)$$

Note that once we have β from the fixed point approach we analyse the entire system of n nodes; hence, the summations in this section will run up to n. Hence the expected amount of data sent in a cycle, will be given by

$$E(L) = \sum_{k=1}^{n} \pi_k E_k L$$

Now, using the renewal reward theorem, the aggregate throughput of the system with n sensor nodes can be seen to be:

$$\Theta(n) = \frac{\sum_{k=1}^{n} \pi_k E_k L}{\sum_{k=1}^{n} \pi_k E_k U}$$

Throughput calculation for a network with a single sensor: For the case of a single node, a much simpler analysis gives the saturation throughput. The average time required for transmission of a packet will be $b_0 + 2\delta + \delta\lceil\frac{\mathcal{T}_{data-ack}}{\delta}\rceil$, where $\mathcal{T}_{data-ack}$ is actual time required for transmission of data and the corresponding acknowledgment, including t_{ack}. Since a new transaction can start at next backoff period boundary only, $\frac{\mathcal{T}_{data-ack}}{\delta}$ has been rounded up. Then,

$$\Theta(1) = \frac{1}{b_0 + 2\delta + \lceil\frac{\mathcal{T}_{data-ack}}{\delta}\rceil\delta}$$

For packet size 10 bytes, $L_{data} = 30$ bytes. Taking into account all headers, t_{ack} and acknowledgment, $\mathcal{T}_{data-ack} =$ 122 symbol times (6.1 backoff periods). Hence $\lceil\mathcal{T}_{data-ack}\rceil$ = 7 backoff periods. With the backoff parameters in the standard, $b_0 = 3.5$ backoff periods. Hence $\Theta(1) = 30$ bytes /12.5 backoff periods = 60 kbps or 250 packets per second.

B. Packet Discard Analysis

It is known from the description of the slotted CSMA/CA algorithm that the backoff cycle for a frame may end with either a successful transmission, a collision or a backoff failure. In case a backoff failure occurs, the frame is discarded without any reattempt. If a frame collides, it is retried for *aMaxFrameRetries*. A frame will collide only if while attempting for this frame, the node finds the channel in first CCA. Let α_{CCA1} be the probability of the channel being in first CCA. Then, using the approach in Section IV-B, the probability that a frame collides, given that it is attempted, can be written as:

$$\alpha_{CCA1} = \frac{\sum_{k=1}^{n} \pi_k E_k R^{(CCA1)}}{\sum_{k=1}^{n} \pi_k E_k U} =: H_{CCA1}(n, \beta)$$

Let $p_{i,j}$ be the probability of a packet being successfully served after it has faced i collisions and is in the j^{th} backoff of current backoff cycle ($0 \le j \le K$). Then $p_{0,0}$ will be the probability that a packet is successfully served. We recall that K and N denote *macMaxCSMABackoffs* and *aMaxFrameRetries* respectively. We see that

$$p_{0,0} = 1 - (\alpha + \alpha_{CCA1}) + \alpha_{CCA1} p_{1,0} + \alpha p_{0,1}$$

and, also

$$p_{0,1} = 1 - (\alpha + \alpha_{CCA1}) + \alpha_{CCA1} p_{1,0} + \alpha p_{0,2}$$

Recursing (using $p_{0,K} = 1 - (\alpha + \alpha_{CCA1}) + \alpha_{CCA1} p_{1,0}$), we get

$$\begin{aligned} p_{0,0} &= \sum_{k=0}^{K} \alpha^k \left(1 - (\alpha + \alpha_{CCA1}) + \alpha_{CCA1} p_{1,0}\right) \\ &= \frac{(1 - (\alpha + \alpha_{CCA1}))(1 - \alpha^{K+1})}{1 - \alpha} \\ &+ \frac{\alpha_{CCA1}(1 - \alpha^{K+1})}{1 - \alpha} p_{1,0} \end{aligned}$$

Similarly

$$\begin{aligned} p_{1,0} &= \frac{(1 - (\alpha + \alpha_{CCA1}))(1 - \alpha^{K+1})}{1 - \alpha} \\ &+ \frac{\alpha_{CCA1}(1 - \alpha^{K+1})}{1 - \alpha} p_{2,0} \end{aligned}$$

Again recursing (with $p_{N+1,0} = 0$), we get

$$\begin{aligned} p_{0,0} &= \sum_{k=0}^{N} \left(\frac{\alpha_{CCA1}(1 - \alpha^{K+1})}{1 - \alpha}\right)^k \\ &\quad \left(\frac{(1 - (\alpha + \alpha_{CCA1}))(1 - \alpha^{K+1})}{1 - \alpha}\right) \end{aligned}$$

Then, the probability of a packet being discarded will be given by

$$P_{discard} = 1 - p_{0,0}$$

Defining $D(n)$ to be the packet discard rate with n nodes, it is easily seen that

$$D(n) = \frac{\Theta(n)}{1 - P_{discard}} P_{discard}$$

C. Numerical Results

For the simulation results ns2 version 2.26 is used, with patches for the IEEE 802.15.4 LR-WPAN code provided by J. Zheng[13]. We have used the source code released on January 1, 2005, with modifications as discussed in Section III-C. Static routing is implemented by using NOAH as the wireless routing agent. This allows us to ensure multihop wireless routing is not used. We work in the 2.45 GHz band, with the PHY data rate 250 Kbps. The simulation scenario consist of n nodes distributed uniformly around a circle of radius 8 meters, with the PAN coordinator at the center. The decoding and the sensing range thresholds of the nodes are set to 20 meters, so that all nodes form a single cell. Nodes start associating with the PAN coordinator one by one at regular intervals of 0.5 seconds. After 5.0 seconds of the last node having started association, CBR traffic is initiated simultaneously from all the nodes. The CBR packet size is kept as 10 bytes to which 20 bytes of IP header, 7 bytes of MAC header and 6 bytes of PHY header are added. All throughput results in packets/sec are provided for the MAC payload. To ensure saturation, the CBR traffic interval is kept very small; each node's buffer receives packets at intervals of 5 ms.

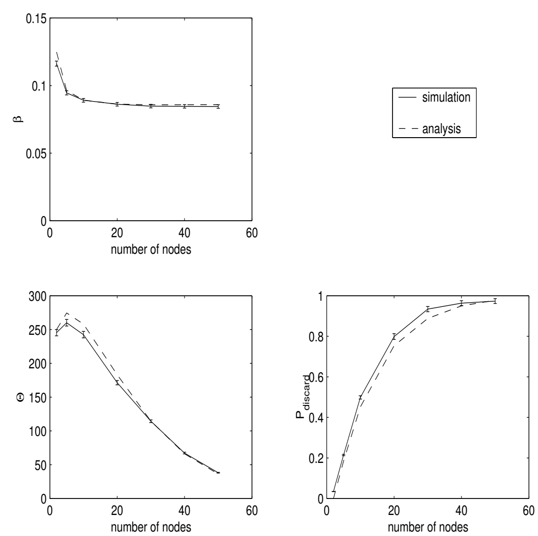

Fig. 16. Analytical and ns2 simulation results for β, $\Theta(n)$ (packets per second), and $P_{discard}$ vs. n. Simulation results are accompanied by 95% confidence intervals.

The conditional attempt rates per node, β, aggregate throughputs, Θ (packets per second), and discard probabilities, $P_{discard}$, obtained through simulation are compared against the analytical results in Figure 16, where 95% confidence intervals are also shown for the simulation results. It can be seen that, even after many modeling simplifications, the fixed point analysis provides an excellent approximation, for a wide range of the number of nodes, n. The results show that β decreases until $n = 10$ and then remains almost constant with increasing n, while the aggregate throughput increases initially but then decreases very sharply with increasing n. The slight increase in Θ with n is because for small n the contention is less and increasing n increases the channel utilisation. The discard probability increases rapidly as the number of nodes increases and approaches approximately 1 as the number of nodes is increased up to 50. We will see later that finite load performance is substantially better, and, in fact, derivable from the saturation performance.

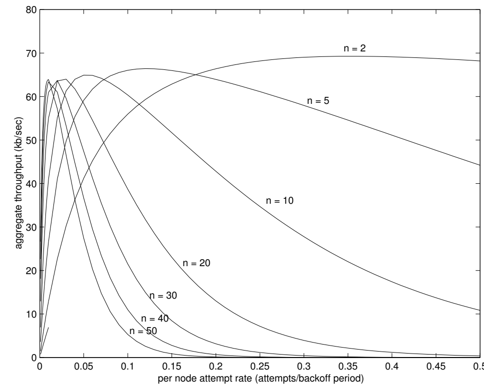

Fig. 17. Aggregate throughput (Θ in Kbps) plots for various n as a function of β, obtained from the stochastic analysis.

VI. Alternate Backoff Parameters

A. Performance and Backoff Parameters

If the expression for $\Theta(\cdot)$ in Section V-A is evaluated with β as a free variable, for various values of β and n, then we obtain the plots in Figure 17; note that here Θ is given in Kbps (60 Kbps = 250 packets per second). For $n \geq 10$, the values of β for which the aggregate throughput Θ peaks are much less than those obtained for the actual system for the default parameters (see Figure 16). Figure 17 also shows that once a node exceeds these attempt rates its throughput starts decreasing as the attempt rate increases. This is the region, in which the network operates with the current set of backoff parameters (see Figure 16). In this regime of operation, to maintain a constant throughput as n increases, β must decrease sharply with n (see Figure 17); this does not occur with the given backoff parameters. It is also observed from Figure 17 that the same throughputs can be obtained with much smaller attempt rates, while working in a region where throughput increases with attempt rate for a fixed number of nodes. The attempt rates, collision probabilities and hence energy expenditure is much less in this region. This also leads to lower discard probabilities. It is also seen that the attempt rate need not decrease significantly in this region to maintain a constant throughput with increasing number of nodes.

We find that simple changes in the backoff parameters can lead us to operate in the desired region. Figures 18 and 19 show the results for two ways of altering the backoff parameters: (i) increasing the backoff multiplier to 3, and (ii) changing macMinBE to 5 and aMaxBE to 7. Figure 18 shows the attempt rate, throughput and discard probability plots after we have increased the backoff multiplier from 2 to 3. Figure 19 is for the case when $macMinBE$ and $aMaxBE$ have been changed to 5 and 7 respectively. These plots show that, with a slight change of a few backoff parameters we are

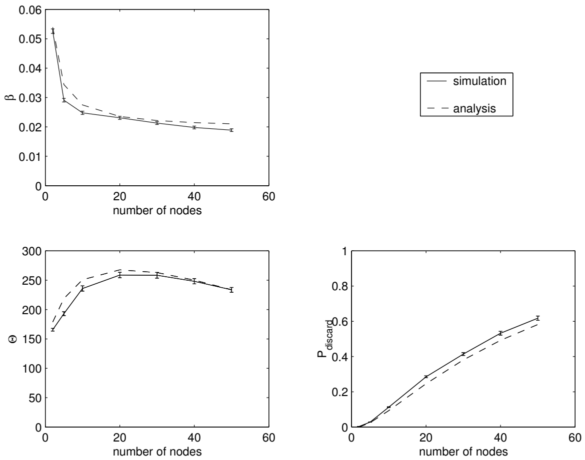

Fig. 18. Analytical and ns2 simulation results for various parameters vs number of nodes. Plots have been obtained using backoff multiplier $p = 3$. Simulation results show 95% confidence intervals.

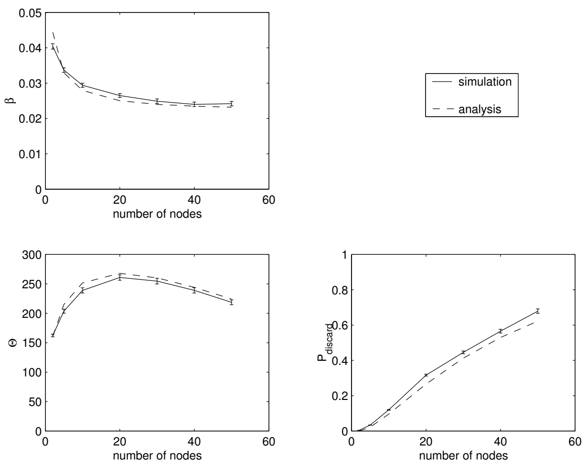

Fig. 19. Analytical and ns2 simulation results for various parameters vs number of nodes. Plots have been obtained with *macMinBE* = 5 and *aMaxBE* = 7. Simulation results show 95% confidence intervals.

able to maintain constant throughput with increasing number of nodes. Discard probabilities are also substantially smaller. The large initial backoffs cause less contention in the case of a large number of nodes, but on the other hand lead to unnecessary wastage of backoff time when number of nodes is small. Hence, we get worse throughputs as compared to those with default parameters for small number of nodes (see Figures 18 and 19). Thus it becomes interesting to consider the possibility of adapting the backoff parameters depending on the number of active nodes.

VII. Analysis with Finite Arrival Rates

Each sensor node receives (generates) packets that have to be delivered to the hub node. We assume that the rate of "arrival" of packets at each sensor node is λ and the arrival processes are independent and Poisson. Define $\Lambda = n\lambda$.

Let ρ denote the fraction of time a sensor node is nonempty and hence contending for the channel. As before, $\Theta(n)$ and $D(n)$ are the aggregate throughputs and discard rates for a network with n saturated nodes. We adopt an approach suggested in [6, Chapter 4]. For fixed n, define

$$\mu(n,\rho) = \sum_{m=1}^{n} \binom{n}{m} \rho^m (1-\rho)^{n-m} (\Theta(m) + D(m))$$

and,

$$\nu(n,\rho) = \sum_{m=1}^{n} \binom{n}{m} \rho^m (1-\rho)^{n-m} \Theta(m)$$

Thus, given ρ, $\mu(\rho)$ is an approximation for the rate at which packets are being removed from the queue, either by successful transmission or discard. Similarly, $\nu(\rho)$ is an approximation for the rate of successful transmission. It can be seen that $\lim_{\rho\to 1} \mu(n,\rho) = \Theta(n) + D(n)$, and $\lim_{\rho\to 1} \nu(n,\rho) = \Theta(n)$. Figure 20 shows plots of $\mu(\rho)$ vs. ρ and $\nu(\rho)$ vs. ρ for several

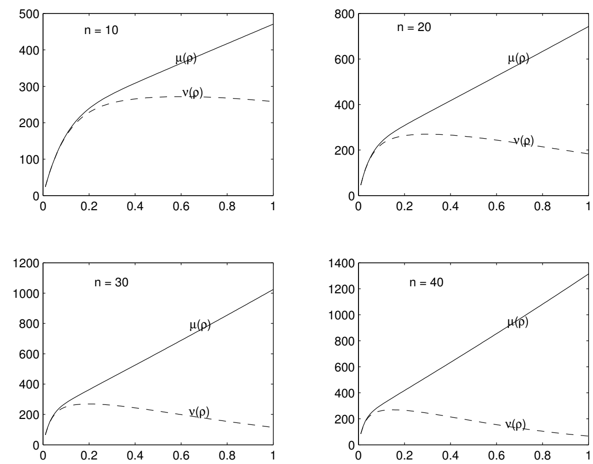

Fig. 20. $\mu(\rho)$ (pkts/sec) and $\nu(\rho)$ (pkts/sec) vs. ρ for $n = 10, 20, 30$ and 40.

values of n. We see that $\mu(\rho)$ monotonically increases with ρ, whereas $\nu(\rho)$ first increases and then decreases with ρ. Hence, for each $\Lambda < \Theta(n) + D(n)$ there will be a unique ρ such that $\mu(\rho) = \Lambda$. We shall take this ρ to be the operating occupancy of a node corresponding to the arrival rate Λ packet per second into each node, i.e., for each n, we take $\rho = \mu^{-1}(\Lambda)$.

Now with the above approximation we can easily obtain various performance measures.

Aggregate throughput: The aggregate throughput of the with n nodes can be obtained as

$$\Phi(n,\Lambda) = \nu(n, \mu^{-1}(\Lambda))$$

Note that, for $\Lambda \geq \Theta(n) + D(n)$, $\Phi(n,\Lambda) = \Theta(n)$.

Attempt rate per node: Corresponding to a finite arrival rate λ, the per node attempt rate of the network (in attempts per backoff period) can be approximately obtained as

$$\beta(n,\Lambda) = \frac{1}{n} \sum_{m=1}^{n} \binom{n}{m} \rho^m (1-\rho)^{n-m} m\beta(m)$$

where, $\beta(m)$ is the per node attempt rate (in attempts per backoff period) for a network with m saturated nodes.

Sojourn time distribution: As per the approximation made, each packet is discarded with probability $P_{discard}$ independently of anything else. Hence we can view each node's queue as having two independent Poisson arrival processes with rates $\lambda P_{discard}$ and $\lambda(1 - P_{discard})$. Assuming an M/M/1 model (i.e., an exponential approximation for the service time) we obtain the data packet mean sojourn time

$$\Delta(n, \Lambda) = \left(\frac{\mu^{-1}(\Lambda)}{1 - \mu^{-1}(\Lambda)} \right) \frac{1}{\lambda}$$

Using an M/G/1 model a more exact sojourn time analysis can be done; however, this requires both first and second moments of the packet service time.

Discard probability: The discard probability is approximately

$$P_{discard} = (\Lambda - \nu(n, \mu^{-1}(\Lambda)))/\Lambda$$

Observations: Figures 21 and 22 show the analytical as well as ns2 simulation results for 20 and 40 node networks. The plots show that the analysis is able to capture the trends of all the performance measures very well in all the cases, and the values of β, ρ, Θ, and $P_{discard}$ are approximated very well. We notice that for small Λ the discard probability is small and the aggregate throughput, Θ, increases with Λ, until Θ peaks and then drops down to the saturation throughput (see Figure16). Notice that with finite load more throughput can be sustained than the saturation throughput. For measurement and control applications, it appears that the discard probability will determine the capacity. The mean delay approximation could be improved by using the M/G/1 mean delay formula if we had the second moment of service time as well.

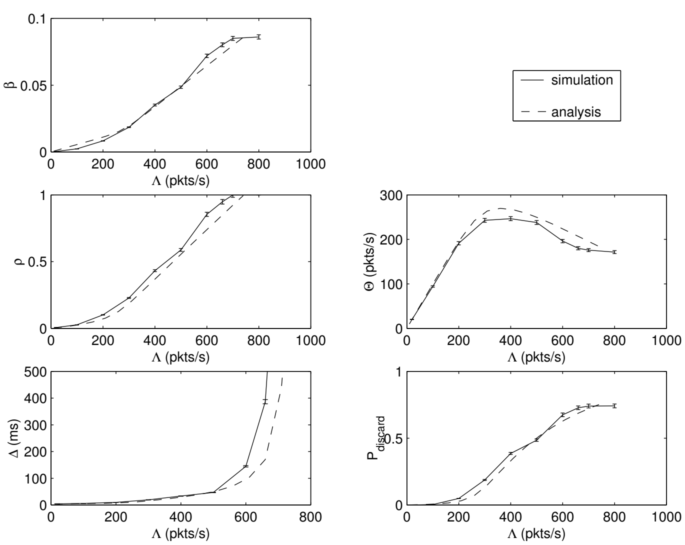

Fig. 21. Analysis and simulation plots for $n = 20$ with default parameters, and Poisson arrivals. Simulation results show 95% confidence intervals.

VIII. Conclusion

We have provided an approximate saturation analysis for a star topology IEEE 802.15.4 network whose function is to make measurements and pass them to the PAN coordinator, and we have validated our results against ns2 simulations. Our analysis is based on identifying a certain Markov renewal process embedded in the system and then using a decoupling approximation that led to a fixed point equation, in much the same spirit as [8] and [3]. Then we have shown how this saturation analysis can be used to develop a finite load analysis, and we found that the approach works remarkably well for this system. In future work we propose to extend our analysis to multihop topologies such as a multilevel star.

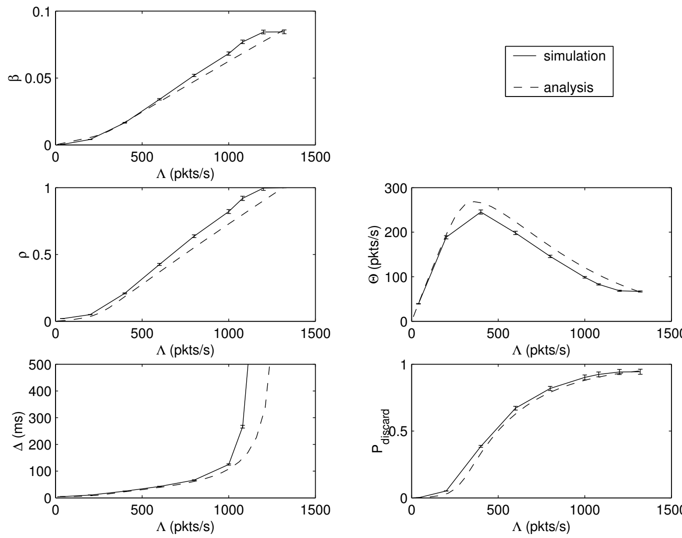

Fig. 22. Analysis and simulation plots for $n = 40$ with default parameters, and Poisson arrivals. Simulation results show 95% confidence intervals.

References

[1] IEEE std. 802.15.4,“Part 15.4:Wireless MAC and PHY specifications for Low-Rate Wireless Personal Are Networks”, *May 2003*.

[2] ZigBee Alliance,“Draft standard: 02130r4ZB-NWK-Network layer specification”, *March, 2003*.

[3] G. Bianchi,“Performance analysis of the IEEE 802.11 Distributed Coordination Function”, *IEEE JSAC, 18(3): 535-547, March 2000*.

[4] Ed Callaway, P. Gorday, L. Hester, J. A. Gutierrez, M. Neave, B. Heile, V. Bahl,“Home Networking with 802.15.4: A developing standard for low-rate wireless personal area networks”, *IEEE Communication Magazine, vol. 40, no. 8, pp. 70-77, August 2002*.

[5] N. Golmie, D. Cypher, O. Rebala,“Performance Analysis of Low Rate Wireless Technologies For Medical Applications”, *IEEE GLOBECOM, 2004*.

[6] Munish Goyal,“Stochastic Control of Wireless Transmissions over Multiaccess Fading Channels”, *Phd. Thesis, IISc, Dec. 2003*.

[7] Patrick Kinney,“ZigBee Technology: Wireless Control that Simply Works”, *Communication Design Conference, October 2003*.

[8] Anurag Kumar, Eitan Altman, Daniele Miorandi and Munish Goyal,“New Insights from a Fixed Point Analysis of Single Cell IEEE 802.11 WLANs”, *IEEE INFOCOM, 2005*.

[9] Jianliang Zheng and Myung J. Lee,“Will IEEE 802.15.4 Make Ubiquitous Networking a Reality?– A Discussion on a Potential Low Power, Low Bit Rate Standard”, *IEEE Communications Magazine, June 2004*.

[10] Gang Lu, Bhaskar Krishnamachari, Cauligi S. Raghavendra,“Performance Evaluation of the IEEE 802.15.4 MAC for Low-Rate Low-Power Wireless Networks”, *IEEE IPCCC, 2004*.

[11] Mathivanan Prabhakaran and Anurag Kumar,“An Analytical Model for a Single Cell IEEE 802.11 WLAN with Open Loop Arrivals”,*submitted, MAY 2005*.

[12] N.F. Timmons and W.G. Scanlon,“Analysis of the Performance of IEEE 802.15.4 for Medical Sensor Body Area Networking”, *IEEE SECON 2004*.

[13] Jianliang Zheng,“[ns] 802.15.4 source code”, http://www-ee.ccny.cuny.edu/zheng/pub.

[14] Jianliang Zheng and Myung J. Lee,“A comprehensive Performance Study of IEEE 802.15.4”, *IEEE Press Book, 2004*.

APPENDIX

A. Analysis of the the Channel Evolution MRP

We see that in the Markov renewal process formulated in Section IV-A, the only feasible possibilities for (U_i, X_i) are $\{(T_{coll} + j\delta, k), j = 2, 3.....J; k = 1, 2, ...n-2; , (T_{data-ack} + 2\delta, n-1), (T_{coll} + (J+1)\delta, n), (\delta, n)\}$. Using the decoupling approximation of Section IV-C, the transition probabilities for the process can be obtained as follows.

A. Transition probabilities for the MRP

1) $X_i = n$ i.e., this cycle starts with any of the nodes being available to attempt.
   a) $(U_{i+1}, X_{i+1}) = (\delta, n)$, if none of the n nodes attempts. Thus,

$$P(U_{i+1} = \delta, X_{i+1} = n | X_i = n) = (1-\beta)^n$$

   b) $(U_{i+1}, X_{i+1}) = (T_{data-ack} + 2\delta, n-1)$, if exactly one of the n nodes attempts. Thus,

$$P(U_{i+1} = T_{data-ack} + 2\delta, X_{i+1} = n-1 | X_i = n) = n\beta(1-\beta)^{(n-1)}$$

   c) $(U_{i+1}, X_{i+1}) = (T_{coll} + j\delta, k_2)$, if exactly $n - k_2 (\geq 2)$ out of n nodes attempt. Further none of the remaining k_2 nodes attempts for $(j-2)$ backoff periods after the collision, and at least one of them attempts in the very next backoff period. Thus, for $1 \leq k_2 \leq n-2, 2 \leq j \leq J$,

$$\begin{aligned} &P(U_{i+1} = T_{coll} + j\delta, X_{i+1} = k_2 | X_i = n) = \\ &\left(\begin{array}{c} n \\ n-k_2 \end{array} \right) \beta^{n-k_2} (1-\beta)^{k_2} ((1-\beta)^{k_2})^{(j-2)} (1-(1-\beta)^{k_2}) \end{aligned}$$

   d) $(U_{i+1}, X_{i+1}) = (T_{coll} + (J+1)\delta, n)$, if $k, 2 \leq k \leq n$, nodes attempt, and none of the remaining $n-k$ nodes attempts for $(J-1)$ consecutive backoff periods after collision in this cycle. Thus,

$$\begin{aligned} &P(U_{i+1} = T_{coll} + (J+1)\delta, X_{i+1} = n | X_i = n) = \\ &\sum_{k=2}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) \beta^k (1-\beta)^{(n-k)} ((1-\beta)^{(n-k)})^{(J-1)} \end{aligned}$$

   It can be checked that

$$\sum_{u,x} P(U_{i+1} = u, X_{i+1} = x | X_i = n) = 1$$

   since

$$\begin{aligned} &(1-\beta)^n + n\beta(1-\beta)^{(n-1)} \\ &+ \sum_{j=2}^{J} \sum_{k_2=1}^{n-2} \left(\begin{array}{c} n \\ n-k_2 \end{array} \right) \beta^{n-k_2} (1-\beta)^{k_2} ((1-\beta)^{k_2})^{(j-2)} (1-(1-\beta)^{k_2}) \\ &+ \sum_{k=2}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) \beta^k (1-\beta)^{(n-k)} ((1-\beta)^{(n-k)})^{(J-1)} = 1 \end{aligned}$$

2) $X_i = n-1$ i.e., this cycle starts with $(n-1)$ of the nodes being available to attempt. This is similar to the case $X_i = n$.
   a) $(U_{i+1}, X_{i+1}) = (\delta, n)$, if none of the $n-1$ nodes attempts. Thus,

$$P(U_{i+1} = \delta, X_{i+1} = n | X_i = n-1) = (1-\beta)^{n-1}$$

   b) $(U_{i+1}, X_{i+1}) = (T_{data-ack} + 2\delta, n-1)$, if exactly one of the $n-1$ nodes attempts. Note that the node that was not included at the start of the cycle would be ready to attempt at the end of the cycle, whether one of the other nodes would not be able to attempt. Thus,

$$P(U_{i+1} = T_{data-ack} + 2\delta, X_{i+1} = n-1 | X_i = n-1) = (n-1)\beta(1-\beta)^{(n-2)}$$

   c) $(U_{i+1}, X_{i+1}) = (T_{coll} + j\delta, k_2)$, if exactly $n - k_2 (\geq 2)$ out of $n-1$ nodes attempt. Further none of the remaining k_2 nodes attempt for $(j-2)$ backoff periods after the collision, and at least one of them attempts in the the very next backoff period. We note here that the node that is not available to attempt at the beginning of the cycle, can attempt after $T_{coll} + 2\delta$ and is included in the calculation. Thus, for $1 \leq k_2 \leq n-2, 2 \leq j \leq J$,

$$P(U_{i+1} = T_{coll} + j\delta, X_{i+1} = k_2 | X_i = n-1) = \left(\begin{array}{c} n-1 \\ n-k_2 \end{array} \right) \beta^{n-k_2} (1-\beta)^{(k_2-1)} ((1-\beta)^{k_2})^{(j-2)} (1-(1-\beta)^{k_2})$$

d) $(U_{i+1}, X_{i+1}) = (T_{coll} + (J+1)\delta, n)$, if $k, 2 \leq k \leq n-1$, nodes attempt, and none of the remaining $n-k$ nodes attempts for $(J-1)$ consecutive backoff periods after collision in this cycle. Thus,

$$P(U_{i+1} = T_{coll} + j\delta, X_{i+1} = n | X_i = n-1) = \sum_{k=2}^{(n-1)} \left(\begin{array}{c} n-1 \\ k \end{array} \right) \beta^k (1-\beta)^{(n-1-k)} ((1-\beta)^{(n-k)})^{(J-1)}$$

As in case 1, it can be checked that

$$\sum_{u,x} P(U_{i+1} = u, X_{i+1} = x | X_i = n-1) = 1$$

since

$$\begin{aligned} &(1-\beta)^{(n-1)} + (n-1)\beta(1-\beta)^{(n-2)} \\ &+ \sum_{j=2}^{J} \sum_{k_2=1}^{n-2} \left(\begin{array}{c} n-1 \\ n-k_2 \end{array} \right) \beta^{n-k_2} (1-\beta)^{k_2} ((1-\beta)^{k_2})^{(j-2)} (1-(1-\beta)^{k_2}) \\ &+ \sum_{k=2}^{(n-1)} \left(\begin{array}{c} n-1 \\ k \end{array} \right) \beta^k (1-\beta)^{(n-1-k)} ((1-\beta)^{(n-k)})^{(J-1)} = 1 \end{aligned}$$

3) Consider $k_1 \in \{1, 2, \ldots, n-2\}, k_2 \in \{1, 2, \ldots, n-2\}, j \in \{2, 3, 4\}$. If $X_i = k_1$ i.e., this cycle starts with k_1 of the nodes being available to attempt. It itself implies that one of them has attempted.

a) $(U_{i+1}, X_{i+1}) \neq (\delta, n)$, because at least one of the k_1 nodes has attempted, and the cycle cannot be an idle one. Thus,

$$P(U_{i+1} = \delta, X_{i+1} = n | X_i = k_1) = 0$$

b) $(U_{i+1}, X_{i+1}) = (T_{data-ack} + 2\delta, n-1)$, if exactly one of the k_1 nodes attempts, given that at least one of them has attempted. Thus,

$$P(U_{i+1} = T_{data-ack} + 2\delta, X_{i+1} = n-1 | X_i = k_1) = \frac{k_1 \beta (1-\beta)^{(k_1-1)}}{1-(1-\beta)^{k_1}}$$

c) $(U_{i+1}, X_{i+1}) = (T_{coll} + j\delta, k_2)$, if exactly $n-k_2 (\geq 2)$ of k_1 contending nodes attempt $(n-k_2 \leq k_1)$. To ensure the following cycle length to be $(2\delta + T_{coll} + (j-2)\delta)$, it is necessary that none of the k_2 nodes attempts during first $(j-2)$ backoff periods after collision but at least one of them attempts in the very next backoff period. The next state cannot be $(T_{coll} + j\delta, k_2)$, if $n-k_2 > k_1$. As we have already discussed availability of less than $n-1$ nodes itself implies that at least one of them has attempted. Thus,

$$P(U_{i+1} = T_{coll} + j\delta, X_{i+1} = k_2 | X_i = k_1) = \begin{cases} \dfrac{\left(\begin{array}{c} k_1 \\ n-k_2 \end{array} \right) \beta^{n-k_2} (1-\beta)^{(k_1+k_2-n)} ((1-\beta)^{(n-k_2)})^{(j-2)} (1-(1-\beta)^{(n-k_2)})}{1-(1-\beta)^{k_1}}, & \text{if } k_1 + k_2 \geq n, k_1 > 1, \\ 0, \text{otherwise.} \end{cases}$$

d) $(U_{i+1}, X_{i+1}) = (T_{coll} + (J+1)\delta, n)$, if some $k, 2 \leq k \leq k_1$, nodes attempt out of the k_1 nodes available to contend. To ensure the cycle length to be $(2\delta + T_{coll} + (J-1)\delta)$, it is necessary that none of the $n-k$ nodes attempts during first $(J-1)$ backoff periods after collision. Thus,

$$P(U_{i+1} = T_{coll} + (J+1)\delta, X_{i+1} = n | X_i = k_1) = \begin{cases} \frac{\sum_{k=2}^{k_1} \binom{k_1}{k} \beta^k (1-\beta)^{(k_1-k)} ((1-\beta)^{(n-k)})^{(J-1)}}{1-(1-\beta)^{k_1}}, \text{if } k_1 > 1, \\ 0, \text{otherwise.} \end{cases}$$

As in cases 1 and 2, again it can be checked that for $k_1 \in \{1, 2, \ldots, n-2\}$,

$$\sum_{u,x} P(U_{i+1} = u, X_{i+1} = x | X_i = k_1) = 1$$

since for $k_1 = 1$,

$$\frac{k_1 \beta (1-\beta)^{(k_1-1)}}{1-(1-\beta)^{k_1}} = 1$$

and for $k_1 > 1$,

$$\frac{k_1 \beta (1-\beta)^{(k_1-1)}}{1-(1-\beta)^{k_1}} + \sum_{k_2=n-k_1}^{n-2} \frac{\binom{k_1}{n-k_2} \beta^{n-k_2} (1-\beta)^{(k_1+k_2-n)} ((1-\beta)^{(n-k_2)})^{(j-2)} (1-(1-\beta)^{(n-k_2)})}{1-(1-\beta)^{k_1}} + \frac{\sum_{k=2}^{k_1} \binom{k_1}{k} \beta^k (1-\beta)^{(k_1-k)} ((1-\beta)^{(n-k)})^{(J-1)}}{1-(1-\beta)^{k_1}} = 1$$

1) Transition Probabilities for the Channel with Single Node:: In case of a channel comprising of single node, although a successful transmission takes only 4 symbol times in the last backoff period, the node cannot attempt for its next packet in the same backoff period. Hence a successful transmission consumes effectively $T_{data-ack} + 3\delta$ time. A backoff period will be an idle cycle, if the node is in backoff and doesnot attempt in that backoff period. There cannot be any collisions. Hence, only feasible possibilities for (U_i, X_i) are $(T_{data-ack} + 3\delta, 1)$ and $(\delta, 1)\}$. Transition probabilities for the Markov renewal process can be obtained as follows.

- $X_i = 1 \forall i$
  1) $(U_{i+1}, X_{i+1}) = (\delta, 1)$, if the node doesnot attempt. Thus,

$$P(U_{i+1} = \delta, X_{i+1} = 1 | X_i = 1) = 1 - \beta$$

  2) $(U_{i+1}, X_{i+1}) = (T_{data_ack} + 3\delta, 1)$, if the node attempts. Thus,

$$P(U_{i+1} = T_{data-ack} + 3\delta, X_{i+1} = 1 | X_i = 1) = \beta$$

B. Transition Probabilities for the Markov Chain $\{X_i, i \geq 0\}$

Let **M** be the transition probability matrix for the one dimensional Markov chain $\{X_i, i \geq 0\}$.
For $k_1, k_2 = 1, 2, 3, ...n-2$;

$$M_{k_1,k_2} = \sum_{j=2}^{J} P(U_{i+1} = T_{coll} + j\delta, X_{i+1} = k_2 | X_i = k_1)$$

$$M_{k_1,n-1} = P(U_{i+1} = T_{data-ack} + 2\delta, X_{i+1} = n-1 | X_i = k_1)$$

$$M_{k_1,n} = P(U_{i+1} = T_{coll} + (J+1)\delta, X_{i+1} = n | X_i = k_1)$$

$$+P(U_{i+1} = \delta, X_{i+1} = n | X_i = k_1)$$

$$M_{n-1,k_2} = \sum_{j=2}^{J} P(U_{i+1} = T_{coll} + j\delta, X_{i+1} = k_2 | X_i = n-1)$$

$$M_{n-1,n-1} = P(U_{i+1} = T_{data-ack} + 2\delta, X_{i+1} = n-1 | X_i = n-1)$$

$$M_{n-1,n} = P(U_{i+1} = T_{coll} + (J+1)\delta, X_{i+1} = n | X_i = n-1)$$

$$+P(U_{i+1} = \delta, X_{i+1} = n | X_i = n-1)$$

$$M_{n,k_2} = \sum_{j=2}^{J} P(U_{i+1} = T_{coll} + j\delta, X_{i+1} = k_2 | X_i = n)$$

$$M_{n,n-1} = P(U_{i+1} = T_{data-ack} + 2\delta, X_{i+1} = n-1 | X_i = n)$$

$$M_{n,n} = P(U_{i+1} = T_{coll} + (J+1)\delta, X_{i+1} = n | X_i = n)$$

$$+P(U_{i+1} = \delta, X_{i+1} = n | X_i = n)$$

C. Conditional Distribution of the Cycle Times

The conditional distribution of the cycle length U_{i+1}, given X_i, can be obtained as follows.
For $k_1 = 1, 2, ...n, j = 2, 3, ..J$

$$P(U_{i+1} = T_{coll} + j\delta | X_i = k_1) = \sum_{k_2=1}^{n-2} P(U_{i+1} = T_{coll} + j\delta, X_{i+1} = k_2 | X_i = k_1)$$

$$P(U_{i+1} = T_{coll} + (J+1)\delta | X_i = k_1) = P(U_{i+1} = T_{coll} + (J+1)\delta, X_{i+1} = n | X_i = k_1)$$

$$P(U_{i+1} = T_{data-ack} + 2\delta | X_i = k_1) = P(U_{i+1} = T_{data-ack} + 2\delta, X_{i+1} = n-1 | X_i = k_1)$$

$$P(U_{i+1} = \delta | X_i = k_1) = P(U_{i+1} = \delta, X_{i+1} = n | X_i = k_1)$$