Modeling the Performance of IoT networks

Sriram Sankaran
Center for Cybersecurity Systems and Networks
Amrita University
Amritapuri, Kollam-690525
Email: srirams@am.amrita.edu

***Abstract*—Internet of Things (IoTs) is gaining increasing significance due to real-time communication and decision making capabilities of sensors integrated into everyday objects. Predicting performance in IoTs is critical for detecting performance bottlenecks, designing optimal sleep/wake-up schedules and application-aware performance tuning. However, performance prediction becomes a significant challenge in IoTs due to varying needs of applications coupled with the resource constrained nature of sensors. In this work, we analyze the impact of factors affecting performance in IoT networks using simulation based models. Further, an analytical framework is developed to model the impact of individual node behavior on overall performance using Markov chains. In particular, we derive steady state transition probabilities of transmit and receive states using protocol execution traces and further utilize them towards predicting per-flow throughput. Our proposed model is generic in that it can be applied across domains. Accuracy of the model is evaluated by comparing the predictions with the actual estimates obtained using simulations.**

I. Introduction

Advances in sensing, computing and communication have changed the Internet for people to Internet of things. IoTs are composed of sensors and actuators embedded into everyday objects that are capable of real-time communication and decision making. In addition, remote monitoring enables IoTs to be deployed in a multitude of application domains such as Smart home, Industrial Automation, Smart Healthcare, Automotive and transportation. The application-driven nature of IoTs leads to numerous challenges which need to be addressed before IoTs are commercially deployed and widely accepted.

IoT applications are typically characterized by communication between embedded sensors, gateway nodes and servers/data centers. The varying needs of applications coupled with the resource constrained nature of sensors make the problem of predicting performance a significant challenge in IoTs. In addition, ever increasing sizes of the datasets generated by IoTs further exacerbates the problem. Thus there exists a need for modeling and analysis of performance in IoTs for different kinds of applications.

IoTs are typically characterized by packets with varying inter-arrival rates that depend on the needs of the applications. Numerous factors affect the performance of applications in IoTs. For instance, mobility of the gateways incurs a significant impact on throughput and round trip times. Similarly, there can be multiple simultaneous applications between different sender-receiver pairs each with different packet sizes and data rates as a result of which power consumption varies across nodes. Thus a need for understanding the impact of these factors on overall performance becomes necessary for IoT networks.

In addition, modeling the performance of IoT networks is challenging due to random behavior of the individual nodes. For instance, routing of data from source to destination impacts the performance of individual nodes along the chosen path. Many routing protocols have been proposed for sensor networks and are applicable within the IoTs. To analyze the impact of routing along the chosen path, stochastic methods are a natural fit towards modeling the performance of the individual flows and the overall network. These methods profile past history of events towards predicting future behavior.

In this work, we analyze factors affecting performance in IoT networks using simulation based models and further develop an analytical framework to model the impact of individual node behavior on performance using Markov chains. In summary, our contributions include:

- Analyzing factors such as gateway mobility and impact of multiple applications on overall end-to-end performance in IoT networks
- Modeling individual node behavior using Markov chains from protocol execution traces and deriving steady-state transition probabilities of transmit and receive states
- Predicting per-flow throughput using transition probabilities and comparing the results with the actual estimates obtained using simulations

II. Related Work

The problem of analyzing factors that contribute towards performance has become increasingly important in computing systems in general and more so in the Internet of things [1]. Numerous efforts have been dedicated to modeling and optimizing performance for a wide range of computing systems. Carroll *et al.* [2] analyzed the power consumed by components in a smartphone. Shnayder *et al.* [3] modeled the energy consumed by different components in low-power embedded devices such as sensors. Agrawal *et al.* [4] modeled the energy consumption of wireless LANs carrying TCP traffic. Murthy *et al.* [5] characterized and modeled the end-to-end performance of indoor powerline networks. Gandhi *et al.* [6] developed queuing models to study the energy-performance trade-offs in datacenters.

Markov Modeling [7] has been utilized in the past for prefetching [8], caching [9], predicting I/O access patterns

[10] [11] and dynamic power management [12] [13]. Mini *et al.* [14] proposed a probabilistic approach to predict energy consumption in wireless sensor networks. Chiasserini *et al.* [15] modeled the energy consumption of sensor networks using 2-state Markov models. Recently, markov modeling has been utilized for disk power management where access patterns for a disk are analyzed and transition probabilities are computed. These transition probabilities are further used to predict future disk idleness for power management.

Performance models have been developed for Wi-Fi based networks [16] [17] [18] using Markov chains. In particular, two-state Markov model consisting of Transmit and Idle states was developed and the impact of interference on the overall performance was analyzed. In one of our own works [19], we analyzed and predicted the power consumed by routing in IoT networks using Markov chains. In contrast to the above works, we analyze factors affecting performance in IoT networks and develop a two-state Markov model consisting of Transmit and Receive states to predict per-flow throughput for IoT applications.

III. Background

A. Internet of Things

The functionality of IoT is illustrated in Figure 1. IoTs are typically organized into three tiers. Tier I contains multitude of embedded devices monitoring objects and their surrounding areas. Tier II represents gateway nodes which receive data from the embedded devices. These gateway nodes are also referred as Edge nodes and are computationally more powerful than the embedded sensors. Tier III contains servers or datacenters which store data received from gateway nodes for processing. Servers or data centers perform complex analytics by developing models of the application behavior using the data received from the gateway nodes.

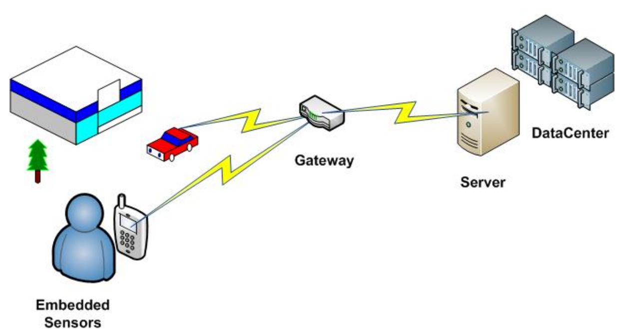

Fig. 1: Internet of Things

Embedded Sensors (Tier I): Typically, sensors in an IoT communicate with each other and the gateway node using the Zigbee protocol which is part of the IEEE 802.15.4 standard commonly referred as Personal Area Networks (PANs). In a Zigbee protocol, sensors are organized in star, ring or mesh based topologies and have considerably lesser transmission power compared to those in Wireless LANs.

Gateway Node (Tier II): Gateway nodes act as an interface between embedded sensors and back-end servers. Typically these nodes are equipped with WLAN (802.11) and WPAN enabled interfaces which have the capability to interact with embedded sensors and back-end servers. In addition, they are computationally powerful and are capable of transmitting data over longer distances.

Back-end Servers (Tier III): Back-end servers are composed of a wide variety of heterogeneous components such as routers and data centers typically communicating through high-bandwidth wired links. These servers are capable of housing and processing significant chunks of real-time data from IoT applications. Typically, big data computing infrastructures such as Hadoop and MapReduce are utilized to perform analytics on IoT applications.

B. Markov Chains

A Markov model is represented by a total number of states n and the transition probability matrix P. These states are used to reflect the context of the system being modeled. For instance, CPU can choose to be in Active mode performing computations or transition to the idle mode after a certain time instant. There can be any number of states and transitions for a system. The probability with which the system transitions from one state to another reflects the actual behavior of the system. A system is said to possess the Markov property if the future state depends only on the current state and the transition probability. The transition probability is a NxN matrix, where N denotes the number of states. Each of the entries in the matrix corresponds to the probability of transitioning from state i to state j.

Transition probability matrix is used to predict next set of observations given the current value. We can evaluate the accuracy by comparing the actual and predicted states. We compute the next state using the following equation.

$$S_1 = S_0 * P$$

where S_0, S_1 and P refer to the initial state, next state and transition probability matrix respectively.

In a regular Markov chain, successive state matrices always approach a unique stationary matrix called the equilibrium or steady state. The steady state can be computed using the following equation.

$$S * P = S$$

where S, and P refer to the stationary matrix and transition probability respectively.

IV. Factors affecting Performance

In this section, we analyze factors affecting performance in IoT networks. In particular, we consider the impact of mobile gateways and multiple traffic sources on performance and energy consumption in IoT networks.

IoTs are characterized by numerous interaction patterns. These patterns vary depending on the needs of the applications.

However we have considered two of the most generic patterns that are applicable across IoT domains. They are termed as Periodic Monitoring and Request-Response based patterns. Below, we provide a description of each of the patterns.

Periodic Monitoring: In a periodic monitoring scenario, sensors typically report observations to gateway nodes such as smartphones. These scenarios are highly prevalent in health care IoTs where patient data is routinely monitored by hospitals or care providers. In this scenario, performance is dependent on the packet size and sampling rate of the application and data rates of the communication channel.

Request-Response: In contrast to a periodic monitoring scenario, request-response based scenarios typically involve gateway nodes requesting data from a specific set of sensors in an on-demand manner. For instance, in smart home based IoTs, users can query a particular room to determine if any of the lights are turned on. In this scenario, performance is dependent on the request rates of the application and number of sensors that are actively in use.

To analyze the impact of performance on IoT networks, we build a simulation model for IoT networks using ns-3 [20] network simulator. We develop periodic monitoring and request response based applications on top of UDP in the simulation model. The proposed simulation model can be used to perform a thorough performance evaluation of IoT networks before deploying them in the actual field.

A. *Effect of Gateway mobility*

We analyze the throughput for a periodic monitoring application running on top of a UDP protocol. In particular, we vary the distance between the sensors and the gateway and measure its corresponding throughput. The results for throughput were obtained using FlowMonitor which is part of ns-3.

Figure 2 contains the results for throughput as a function of distance. The results indicate that as distance increases, throughput of the IoT network decreases. This may be due to the increase in the time it takes to receive the packet resulting in decreased throughput.

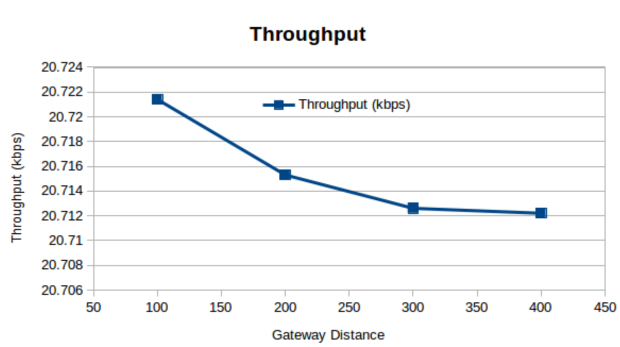

Fig. 2: Throughput

In addition to throughput, we developed a request-response based scenario using ns-3 for measuring the round-trip time (RTT). In a request-response based scenario, a 10-byte request packet is generated every second by the gateways and addressed to the sensors over a multi-hop network and a 1024-byte response is returned to the gateways.

Figure 3 contains the results for RTT for request-response based applications. In particular, we increased the distance from the gateway to the querying nodes by 100m and measured the RTT for request-response based applications. The results indicate that RTT increases with increase in the distance from the gateway nodes. In addition, we observed minimal variations in the RTTs across time intervals for each of the distances.

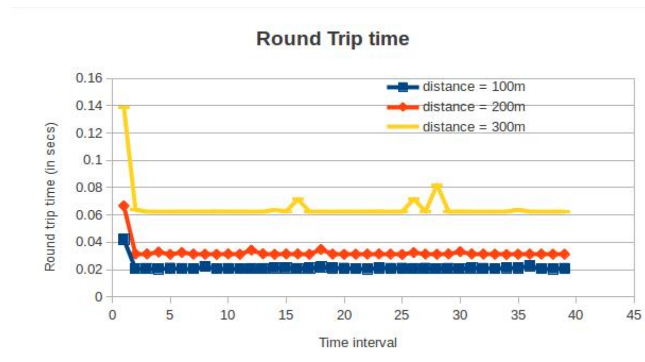

Fig. 3: Round-trip time

B. *Impact of multiple applications*

In addition to single gateways, we envision that multiple gateways can be utilized to query multitude of sensors depending on the application requirements. As a result, there can be varying number of traffic sources each with different packet sizes and data rates. For instance, in a smart home based IoT, while a user queries the room to determine if the lights are turned on, another application would be to query outside temperature so as to adjust the heating conditions in the house. Thus multiple applications are common within the context of IoT networks.

Figure 4 contains the results for throughput as a function of multiple applications. In particular, we varied the number of traffic sources and measured the corresponding throughput. It is evident from the figure that as the number of traffic sources increases, throughput decreases. This may be due to the contention between applications thus resulting in decreased throughput. In addition, we also observed the packet delivery ratio to drop to 66% as the number of traffic sources increased.

C. *Power Consumption*

In addition to throughput and round-trip time, we measured the power consumption of the gateway nodes using a simple battery based energy model developed using ns-3. The proposed energy model operates with a battery capacity of 9720 watt-seconds and consumes about 0.35W, 0.025W and 0.2W of idle, receive and transmit power respectively.

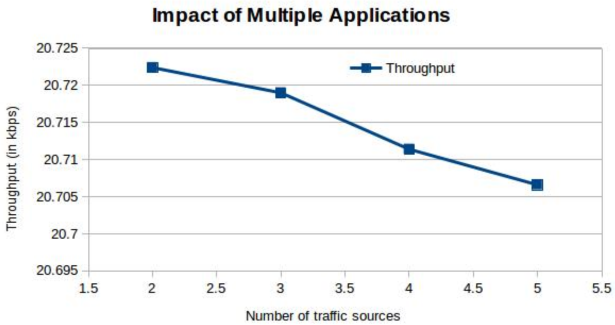

Fig. 4: Impact of multiple applications

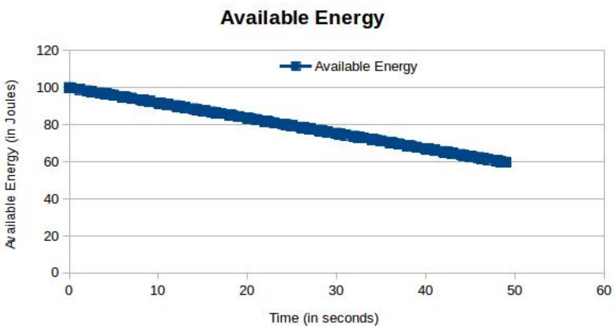

Fig. 5: Available Energy of the Gateway node

Since sensors periodically disseminate data to the gateway nodes,we measure the rate at which battery drains at the gateway node using the energy model. Figure 5 contains the results for available energy at the gateway node as a function of time interval. From the figure, it is evident that the battery drain is strictly linear for the gateway node. Our proposed energy model is generic and can be applied to devices with varying computational capabilities.

V. Model Construction

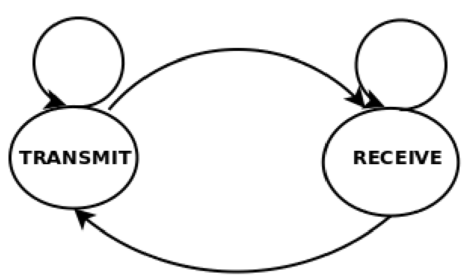

Fig. 6: Markov Model for IoT

In this section, we describe the construction of the 2-state Markov model used to predict per-flow throughput for IoT networks. In IoT applications, each node can choose to transmit or receive depending on node characteristics. For instance, nodes closer to gateways can receive as well as transmit packets more often than those farther from them. Based on these insights, we construct the Markov model for each of the nodes in the network. Each node in the Markov model can be in 2 different states, "Transmit" and "Receive". This means that nodes transmit for a certain period of time or transition to the "Receive" mode after a certain time instant. Further, it is important to note that the sink nodes become absorbing states since packets do not travel beyond these nodes. Figure 6 pictorially describes the Markov chain.

State	Transmit	Receive
Transmit	0.54	0.46
Receive	0.55	0.45

State	Transmit	Receive
Transmit	0	0
Receive	0	1

TABLE I: Transition Probabilities for Forwarder and Sink nodes

We construct the model using execution traces from the ns-3 network simulator. In particular, we observe the time series sequence of routing events in the trace files and wrote Perl scripts to extract the events for each node. The events extracted for each node contain a sequence of transmission and reception messages which vary for individual nodes depending on whether nodes act as sources, forwarders or sinks. Further these transition probabilities were used to generate the steady-state matrices for each of the nodes in the flow. Table I contains the results for the transition probabilities of forwarder and sink nodes for a single flow in the network.

The transition probabilities for the forwarder node indicate that when a node starts with the "Transmit" state, it is most likely to transition to the same state since the probability of transitioning to the same state is higher. Similarly, when a node starts with the "Receive" state, the probability of transitioning to the "Transmit" state is more likely than transitioning to the same state. However, transition probabilities for the sink node indicate that it always receives packets. This may not be realistic since sink nodes can also generate or forward packets destined for other nodes. We planned to develop models for such scenarios in our future work.

VI. Throughput Prediction

In this section, we use steady state transition probabilities derived for each node towards predicting per-flow throughput for IoT applications. We assume that sources nodes discover routes to the destination nodes through pro-active or reactive routing protocols. In addition, nodes that generate packets can also act as forwarders forwarding data destined for other nodes. However, destination nodes store received packets and do not forward them.

Let us assume that there are N nodes in the constructed path which includes the source and the destination nodes. Our intuition is to profile the sequence of transmitted and received packets on the $(N-1)^{th}$ node and construct a Markov

TABLE II: Simulation Parameters

Parameter	Value
Number of nodes	100
Seed	1-100
Topology	100*100
MAC/PHY	802.15.4
Channel	Wireless Channel
Radio Propagation	Two Ray Ground
Beacon Interval	10s
Constant Bit Rate Interval	2s
Bandwidth	0.1Mb

model so as to obtain the steady state transition probabilities for that node. Assuming that the $(N-1)^{th}$ node is more likely to transmit to N^{th} node would enable us to predict the throughput. We rely on this intuition towards predicting per-flow throughput for IoT applications.

We obtain the sequence of transmitted and received packets on the $(N-1)^{th}$ node for the entire simulation and derive the steady state transition probabilities $[P_t \ \ P_r]$. Our simulation parameters are outlined in Table II. We obtained the values for steady state transition probabilities, P_t and P_r to be 0.54 and 0.46 respectively.

We compute the throughput, T from the model using the following equation.

$$T = (P_t^{N-1} * P_{recv}^{N} * numBytes * 8)/((T_{recv} - T_{sent}) * 1024)$$

where P_t^{N-1}, P_{recv}^{N}, $numBytes$, T_{recv} and T_{sent} refer to steady-state probability of transmission of $(N-1)^{th}$ node and probability of reception of N^{th} node, number of bytes received, time at which packets are received and sent in each slot respectively. Since the N^{th} node (sink node) remains in receive mode, its probability of reception P_{recv}^{N} = 1.

For each slot, we predicted the throughput by plugging in the values for P_t^{N-1} and P_{recv}^{N} from the model and the remaining values were observed through simulations. In addition, we computed the actual values for the throughput from the simulation for each slot and compared the actual estimates with the predicted observations towards evaluating the accuracy of the model.

Figure 7 contains the results for model accuracy evaluation. The accuracy of the model is primarily dependent on P_t, which is the steady state probability of transmission from $(N-1)^{th}$ node. When the value of P_t becomes closer to 1, the accuracy of the model increases. It is important to note that the value of P_t is dependent on the sequence of the transmission and reception messages as observed for the $(N-1)^{th}$ node. It is evident from the figure that the predicted values for per-flow throughput lie closer to the actual observations. We propose to improve the accuracy of the model and predict other performance metrics such as delay as part of our future work.

Model Generalization:

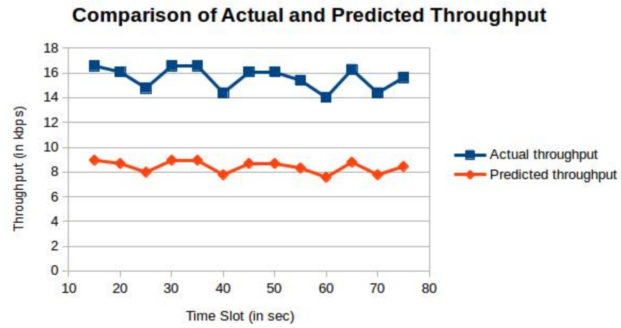

Fig. 7: Model accuracy

Since massive amount of nodes are envisioned to be deployed for IoTs, it is necessary that the model scales to required number of nodes for throughput prediction. Our model for per-flow based throughput prediction can be extended to include required number of flows so as to obtain average per-flow throughput. Total throughput T_{total} can be computed using the following equation.

$$T_{total} = \sum_{i=1}^{n} (P_t^i * P_{recv}^i * numBytes_i * 8)/((T_{recv}^i - T_{sent}^i) * 1024)$$

where n is the number of flows. It is important to note that parameters required for computing throughput vary for each flow. In particular, steady-state transmission and reception probabilities along with the duration of the time slot differ depending on the time varying sequence of transmission and reception patterns contained on the flow. Further, average per-flow throughput, T_{avg} can be computed using the following equation.

$$T_{avg} = T_{total}/n$$

Thus, depending on the needs of applications, throughput estimates obtained from the model can be used for optimizing the performance of IoT networks.

VII. Discussion

Impact of Applications:

We modeled the impact of individual node behavior on overall performance for generic applications. However, emerging standards like IEEE 1905.1 [21] include support for multiple networks such as Ethernet, Wifi, Power Line Communications (PLC) and co-axial cables interoperating with each other via IEEE 1905.1 abstraction layer. Modeling end-to-end performance becomes a significant challenge in such scenarios since data travels across multiple networks each with different operating requirements. We plan to investigate these ideas as part of future work.

Power Management:

Power Management is one of the foremost concerns in IoT networks due to the resource-constrained nature of the sensors.

Network simulation tools typically do not model the impact of CPU which also contributes towards power consumption. Modeling CPU power consumption enables us to understand the computation-communication trade-offs that vary for different applications. One of the approaches to model CPU activity is to develop a utilization based model to predict power consumption. However, modeling CPU activity becomes a significant research challenge when multiple applications are concurrently in operation.

VIII. Conclusion

In this paper, we modeled and analyzed factors affecting performance in IoT Networks. In particular, we developed a simulation-based model to evaluate the impact of factors such as gateway mobility and multiple applications on overall performance. In addition, an analytical framework is developed using Markov chains to model the impact of individual node behavior towards predicting performance. We derived steady-state transition probabilities from protocol execution traces and further utilized them towards predicting per-flow throughput. Our model is generic in that it can be applied across domains. We evaluated the accuracy of the model by comparing the predictions with the actual estimates obtained using simulations. The throughput estimates obtained from the model can be utilized for detecting performance bottlenecks, designing optimal sleep/wake-up schedules and application-aware performance tuning.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[2] A. Carroll and G. Heiser, "An analysis of power consumption in a smartphone." in *USENIX annual technical conference*, 2010, pp. 271–285.

[3] V. Shnayder, M. Hempstead, B. Chen, G. W. Allen, and M. Welsh, "Simulating the power consumption of large-scale sensor network applications," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*, ser. SenSys '04. ACM, 2004, pp. 188–200.

[4] P. Agrawal, A. Kumar, J. Kuri, M. K. Panda, V. Navda, R. Ramjee, and V. N. Padmanabhani, "Analytical models for energy consumption in infrastructure wlan stas carrying tcp traffic," in *Communication Systems and Networks (COMSNETS), 2010 Second International Conference on*, Jan 2010, pp. 1–10.

[5] R. Murthy, J. Padhye, R. Chandra, A. R. Chowdhury, and M. Welsh, "Characterizing the end-to-end performance of indoor powerline networks," Harvard University, Tech. Rep., 2008.

[6] A. Gandhi, M. Harchol-Balter, R. Das, and C. Lefurgy, "Optimal power allocation in server farms," in *Proceedings of the Eleventh International Joint Conference on Measurement and Modeling of Computer Systems*, ser. SIGMETRICS '09. New York, NY, USA: ACM, 2009, pp. 157–168.

[7] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, Feb 1989.

[8] D. Joseph and D. Grunwald, "Prefetching using markov predictors," in *Proceedings of the 24th Annual International Symposium on Computer Architecture*, ser. ISCA '97. New York, NY, USA: ACM, 1997, pp. 252–263.

[9] K. Li, W. Qu, H. Shen, D. Wu, and T. Nanya, "Two cache replacement algorithms based on association rules and markov models," in *Semantics, Knowledge and Grid, 2005. SKG '05. First International Conference on*, Nov 2005, pp. 28–28.

[10] J. Oly and D. A. Reed, "Markov model prediction of i/o requests for scientific applications," in *Proceedings of the 16th International Conference on Supercomputing*, ser. ICS '02. New York, NY, USA: ACM, 2002, pp. 147–155.

[11] T. M. Madhyastha and D. A. Reed, "Input/output access pattern classification using hidden markov models," in *Proceedings of the Fifth Workshop on I/O in Parallel and Distributed Systems*, ser. IOPADS '97. New York, NY, USA: ACM, 1997, pp. 57–67.

[12] Q. Qiu and M. Pedram, "Dynamic power management based on continuous-time markov decision processes," in *Proceedings of the 36th Annual ACM/IEEE Design Automation Conference*, ser. DAC '99. New York, NY, USA: ACM, 1999, pp. 555–561.

[13] T. Simunic, L. Benini, P. Glynn, and G. De Micheli, "Dynamic power management for portable systems," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '00. New York, NY, USA: ACM, 2000, pp. 11–19.

[14] R. A. Mini, B. Nath, and A. A. Loureiro, "A probabilistic approach to predict the energy consumption in wireless sensor networks," in *IV Workshop de Comunicao sem Fio e Computao Mvel*, 2002, pp. 23–25.

[15] C. F. Chiasserini and M. Garetto, "Modeling the performance of wireless sensor networks," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 1, March 2004, p. 231.

[16] R. Boorstyn, A. Kershenbaum, B. Maglaris, and V. Sahin, "Throughput analysis in multihop csma packet radio networks," *IEEE Transactions on Communications*, vol. 35, no. 3, pp. 267–274, Mar 1987.

[17] S. C. Liew, C. Kai, J. Leung, and B. Wong, "Back-of-the-envelope computation of throughput distributions in csma wireless networks," in *2009 IEEE International Conference on Communications*, June 2009, pp. 1–6.

[18] L. Jiang and J. Walrand, "A distributed csma algorithm for throughput and utility maximization in wireless networks," in *Communication, Control, and Computing, 2008 46th Annual Allerton Conference on*, Sept 2008, pp. 1511–1519.

[19] S. Sankaran and R. Sridhar, "Modeling and analysis of routing in iot networks," in *2015 International Conference on Computing and Network Communications (CoCoNet)*, Dec 2015, pp. 649–655.

[20] "ns-3 network simulator," http://www.nsnam.org, november 2010.

[21] "Ieee standard for a convergent digital home network for heterogeneous technologies," *IEEE Std 1905.1-2013*, pp. 1–93, April 2013.