

## Research Article

# Stochastic Modeling and Analysis with Energy Optimization for Wireless Sensor Networks

**DongHong Xu and Ke Wang**

*School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China*

Correspondence should be addressed to DongHong Xu; [xudhxuzihan@163.com](mailto:xudhxuzihan@163.com)

Received 12 March 2014; Accepted 22 April 2014; Published 12 May 2014

Academic Editor: Shanshan Zhao

Copyright © 2014 D. Xu and K. Wang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Due to the battery limitation of wireless sensor network (WSN), there is imperative requirement of energy saving and optimization in practical application of WSN. In order to optimize the energy of WSN, in this paper, a stochastic energy model for WSN is proposed. The model is based on the continuous time Markovian decision processes (CMDP) machine and spans both energy consumption and ensued transition costs. We use PRISM to analyze stochastic CMDP strategies of sensor devices as the formal framework. Simulation experiment is carried out to evaluate the model with given performance criteria. The result shows that the given model can effectively balance energy optimization and perform with high efficiency in WSN.

## 1. Introduction

Wireless sensor networks (WSNs) can collect information and monitor situations in a variety of scenarios including coal mines, battlefields, forests, and farmlands [1–3]. Compared to other self-organized networks, the nodes in WSNs are generally powered by small inexpensive batteries. Each operation, calculation, and intercommunication consume the nodes' energy. Once some nodes, especially those crucial nodes like the cluster head, run out of energy, they will directly influence the connection of WSNs and the monitoring process. To extend the network's lifetime of power constrained wireless sensor networks for a long period of time, their energy consumption should be well managed by optimal routing designs.

Those nodes have the common attractive traits that they have small size, light weight, high mobility, and smart usage, and the most important feature is that they are all battery aware devices. As we move from the wired to wireless sensor network domain, one of the common problems in those battery-operated devices is the availability and cost of the energy. Power savings in these computing devices become more crucial and this constrains the budgets of the data collecting and computing. Google engineers, maintaining thousands of servers, warned that if power consumption continues to grow, power costs can easily overtake hardware costs

by a large margin [4]. Each of us has experienced the event that the battery of our laptop or mobile phone is depleted. The issue is even more serious in autonomous, distributed devices such as sensor networks where the charging of batteries on running is difficult or impossible. Finally, energy dissipation causes thermal problems. Most of the energy consumed by a system is converted into heat, resulting in wear and reduced reliability of hardware and software components.

The research community, hardware manufacturers, and OS designers [5] already found positive solutions to extend the battery life of mobile nodes in WSNs at different levels such as hardware [6], operating system, wireless technologies, and applications [7]. However, these efforts are limited by the heavy layering existing on WSNs platforms that makes difficult exploiting cross layer optimizations, which might be fairly straightforward otherwise. One of the main reasons behind this limitation is a complex business ecosystem in which multiple players (e.g. mobile network providers, content providers, cloud service Providers, hardware manufacturers and OS vendors) compete to retain their share of the mobile business. Most of WSNs have power monitor to allow hardware and software components to dynamically adjust power consumption based on required function and services quantity. However, each kind of approach to power savings has its own limitations. Much of the previous work on dynamic power management has been based on ad hoc

techniques, such as the use of regression equations, interpolation, or learning based methods. Stochastic approaches [8, 9] tend to be more formal in the sense that they are based on mathematical models which make precise assumptions about the probabilistic characteristics, for example, when service requests arrive at a device and how long the device takes to respond to these requests. Validation and analysis of the stochastic DPM [10] schemes, however, are less formal and evaluation is usually being carried out with simulation techniques which are time consuming and often not completely reliable.

The material in the paper is organized as follows. We introduce the models of three different states and transition paths in Section 2. The general CMDP framework and the PRISM tool are presented in Section 3. Simulation example and experiment numerical analysis results are showed to be applying PRISM in Section 4. Finally, the conclusion and future work are described in Section 5.

## 2. The Energy Consumption Model

In this section, we present a stochastic energy consumption model. We prefer a stochastic model; the mean sojourn time for each state is known in advance, but the delay is not deterministic. For example, the sleep state is actually distributed with exponential distribution using the rate for sleep. The same principle is applied for idle and active states as well. We believe that such a stochastic model is more convenient and more realistic than using deterministic delay, since the philosophy of WSN can be described as “finish your job and go to sleep as soon as possible.” Thus, each process should go to sleep as long as it is done.

In the paper, the energy consumption model is majorly fixed in the active state. Let  $r$  be the energy consumption rate, measured in energy units per time unit in the active state. From a practical point of view, it is instructive to study stochastic settings where the length of idle periods is governed by probability distributions [11]. In the following, we assume that the distribution is continuous-time Markov distribution (CTMD). The node energy consumption devices in general have three states: active, idle, and sleep. Each state has two types of energy consumption: state consumption and state transition consumption (Figure 1). We model the states as a stochastic Markov chain process  $\{S_{act}^e(t)\}$ . In this set,  $S$  stands for states of the sensor nodes state. In the paper, three states are considered,  $S_a$ ,  $S_i$ , and  $S_s$ , and are called action states. The set of actions is denoted by  $ACT$ . An action  $act \in ACT$  is the number of time slots the device waits until making the next detecting. When the action occurs, the device passes into one of the three states. These three action events correspond to a device passing to the states “in active,” “in idle,” and “in sleep.” “ $t$ ” is the time for illustrating that one action is chosen at every unit of time. When the action state  $S_{act}^e$  occurs at the moment  $t$ , it is represented by  $S_{act}^e(t)$ . “ $e$ ” is the energy consumption, measured by the states energy consumption and transition energy consumption in one state. Let  $r$  be the energy consumption rate, measured in energy units per time unit, in the active state. From a practical point

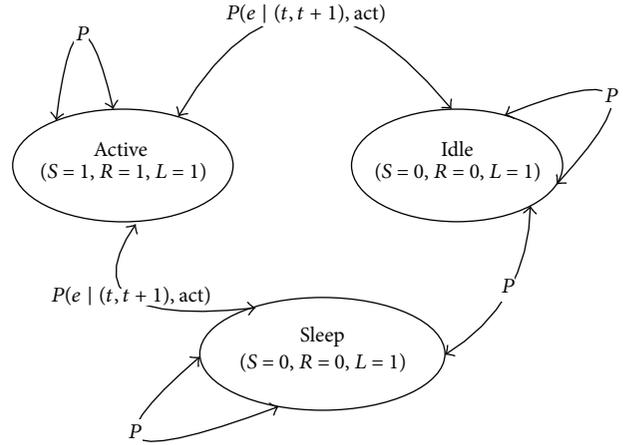


FIGURE 1: Illustration of the states and the states transition path of general WSNs.

of view, it is instructive to study stochastic settings where the length of idle periods is governed by probability distributions. “act” is the action for specifying the action state. Each action state has three type actions, listening, sending, and receiving, and each action has two values: 0 and 1. The expression  $S_{act}^e(t)$  expresses the state that is staying.

At the beginning of the time interval  $t$ , the variable describing these factors is set to  $S_{act}^e(t)$  and changes during the next time step to  $S_{act}^e(t+1)$  with the transition probability  $\rho(S_{act}^e(t+1) | S_a^e(t))$ . This distribution depends on state  $S_{act}^e(t)$  and hence captures the variations in accuracy with changing actions. In real activities, the transition may happen due to the random arrival of signal. The form of the distribution is assumed to be a two dimensional Gaussian distribution centered at  $S_{act}^e(t)$ :

$$\begin{aligned} & \rho\left(S_{act}^e(t+1) | S_a^e(t)\right) \\ &= \frac{1}{\sigma_{s(t)}^2 \sqrt{2\pi}} \exp\left(-\frac{|S(t+1) - s(t)|^2}{\sigma_{s(t)}^2}\right) \in [0, 1]. \end{aligned} \quad (1)$$

In general, we assume the probability  $\rho$  is not known in advance, but we can compute the transition probability  $\rho$  according to the condition probability  $\rho(S_{t+1} | S_t) = (\rho(S_t | S_{t+1})\rho(S_{t+1})) / (\sum_{t=1}^n \rho(S_t | S_{t+1})\rho(S_{t+1}))$  to calculate the next state reaching probability. In fact, from the daily usage of the WSN nodes, we cannot give the  $\rho(S_t | S_{t+1})$  current state probability, so it is difficult to compute the next state probability. Given the fact that we can detect state  $x$  at some time  $t_i$  and the next detects state  $y$  at time  $t_j$  ( $t_i, t_j \in t$  and  $t_i < t_j$ ) easily, the probability of being in state  $z$  at time  $t$  is given by

$$\begin{aligned} & \rho[S_t = z | S_i = x, S_j = y] \\ &= \frac{\rho[S_i = x, S_j = y | S_t = z]}{\rho[S_j = y | S_i = x]} \end{aligned}$$

$$\begin{aligned}
&= \frac{\rho [S_t = z | S_i = x] \cdot \rho [S_j = y | S_t = z]}{\rho [S_j = y | S_i = x]} \\
&= \frac{\rho_{xz}^{t-i} \cdot \rho_{zy}^{t-j}}{\rho_{xy}^{i-j}}.
\end{aligned} \tag{2}$$

In order to gain the “most likely” state, the quantity given in the above formula needs to be maximized:

$$\begin{aligned}
S'_t &= \max_z \{ \rho [S_t = z | S_i = x, S_j = y] \} \iff \\
S'_t &= \max_z \left\{ \frac{\rho_{xz}^{t-i} \cdot \rho_{zy}^{t-j}}{\rho_{xy}^{i-j}} \right\}.
\end{aligned} \tag{3}$$

$R_a(S, S')$  is the reward associated with the transition  $S$  to  $S'$ . When the action act was chosen,  $R_a(S, S')$  was produced at once. In the model, the reward will reflect the reward energy for electricity consumption associated with the transition. We note that these rewards are meant to achieve a desired performance on the node and are therefore tunable to achieve desired behaviors. For example, if request is less valuable than data synchronization, the reward for request can be lowered with respect to data reward. We propose to use the CMDP approach to better handle the dynamics of the system because WSN's behavior is real stochastic. Reward is positive and it is usually interpreted as income and otherwise as cost. Actions are listening, sending, and receiving.

For short-term, small-scale evaluations a wireless sensor network with regular nodes layout is implemented. 19 sites, each with 3 sectors, 10 MHz bandwidth operating at 2.1 GHz carrier frequency, are assumed. Moreover,  $2 \times 2$  MIMO transmission with adaptive rank adaption is assumed. The intersite distance (ISD) for the dense urban and urban environments is set to 500 m, whereas the ISD for suburban and rural areas is set to 1732 m. The users are uniformly distributed, with population densities corresponding to the respective deployment scenarios. The simulation parameters are taken from 3GPP specifications [4]. Two cases are simulated: (a) base stations without sleep mode and (b) base stations with microsleep during idle transmit intervals, that is, neither data nor control signals are transmitted.

The power per area unit  $P/A$ , expressed in [kW/km<sup>2</sup>], with an ISD of 500 m corresponds to a coverage area of 0.2165 km<sup>2</sup> per site. The power consumption is around 4.15 kW/km<sup>2</sup> at low loads, and it can reach 5.1 kW/km<sup>2</sup> at high loads. For the network with microsleep capable base stations, the corresponding figures are 3.5 kW/km<sup>2</sup> at low loads and can reach 5 kW/km<sup>2</sup> at high loads. For comparison, in an empty network when only control channels are transmitted, but no user data, the power consumption equals 885 W per site, which corresponds to a power per area unit of 4.1 kW/km<sup>2</sup>. In the (hypothetical) extreme case, when nothing at all is transmitted (i.e., no data and no control channels) so that the RF output power is 0W, we obtain  $P/A = 3.3$  kW/km<sup>2</sup>. The power consumption per area unit for suburban and rural areas is substantially lower, which

TABLE 1: Average power consumption (W) and service times (ms) for each power state.

	Sleep	Idle	Active
Power consumption (W)	0.2	1.3	2.8
Service time (ms)	0	0	1

TABLE 2: Average state transitions consumption results (W).

	Active	Idle	Sleep
Active	0	0.3	0.2
Idle	0.8	0	0.1
Sleep	1.2	0.05	0

is due to the increased ISD of 1732 m, which corresponds to a coverage area of 2.6 km<sup>2</sup> [12, 13]. However, the system throughput per area unit decreases accordingly, due to the increased site coverage area. Average power consumption (W) and service times (ms) for each power state and average state transitions consumption results (W) are separated and depicted in Tables 1 and 2.

For the analysis of the energy consuming of the sensor node, we limit each state has a finite queue length, say size of  $Q$ , to store the data it generated or forwarded from other state for relaying purpose.

### 3. CMDP Model and Performance Characteristics

The main challenge of modeling is to manage its complexity in terms of the number of states, the number of actions, and the time horizon. This is important because ultimately the optimal energy consuming, typically in the form of a precomputed table, will itself be running on a resource-constrained device (i.e., a sensor node). In order to give a formal model to be feasible for model checking, it should have an explicit state space. We present the situation with numerical values and examples. In general, the model consists of services requestor (SR) to generate the tasks to be serviced, a single service provider (SP) to supply the required services, and a service queue (SQ) to store the tasks waiting for services. Each service may consist of numbers of sensors states. For example, there are five sensors built into most models of the WSN node. Without these sensors, none of the devices would be what we know them as today. They are proximity sensor, motion sensor/accelerometer, ambient light sensor, moisture sensor, and gyroscope. The total number of modules: #modules =  $(N + 1) * M$ .  $M$  services,  $N$  sensors in each service, in the real case:  $M = 3$ ,  $N = 5$ , #modules = 15. For example, assuming that each module has 3 states, the whole model has roughly 3 power 15 = 14348907, and the transition is 3 power 15 \* 16 - 1 = 229582511.

Here we adopt the continuous-time Markovian decision processes approach to the problem of interest. CMDP is a commonly used technique in the field of performance and dependability analysis of sensor and especially of real-time communication system. A CMDP is defined as a 5-tuple,

$(S, S_0, Act, P, R)$ , where  $S$  is the finite set of states, which is denoted as  $S = \{s_0, s_1, s_2, \dots, s_N\}$ , and the state at time  $t$  is denoted as  $s_t$ , a direct product of the handset states set  $S(D)$ , and the indirect states set  $S(i)$ ,  $S = S(D) \otimes @S(i)$ ,  $S_0 \in S$  is the initial state;  $Act$  is a finite set of actions.  $Act(s)$  is the finite set of actions available from state  $s \in S$ . A typical system with a limited battery may have around 800 states and 1600 actions. In practice, the real tool performance is often much better than this worst case analysis, especially for sparse linear programs.  $Act$  is a finite set of actions. Within the framework, the actions are constrained in the listening, receiving, and sending.  $P$  is the transition probability between states that action  $act$  chosen while in state  $s$  at time  $t$  will lead to state  $s'$  at time  $t + 1$ . The probabilistic description of the transition allows accounting for stochastic nature of the energy fluctuations as well as for the randomness in the dynamics of the sensor node.  $R$  is the reward associated with the transition  $s \rightarrow s'$  when the action  $act$  was chosen. In the models, the reward will reflect the cost for energy consumption associated with the transition.

PRISM [14] is a probabilistic model checker, a tool for formal modeling and analysis of systems that exhibit random or probabilistic behavior. It supports a wide range of probabilistic models and has been used to analyze systems from many different application domains, including communication and multimedia protocols, randomized distributed algorithms, security protocols, biological systems, and many others. It supports analysis of several types of probabilistic models: discrete-time Markov chains (DTMCs), continuous-time Markov chains (CTMCs), Markov decision processes (MDPs), probabilistic automata, and probabilistic timed automata. In PRISM, models are described as the parallel composition of a set of modules, with each module containing a set of commands describing transitions. Each command consists of optional action and Boolean formulae over simple arithmetic expressions of variables and constants for antecedent and consequent. Variables can be local with respect to a module or global with respect to the whole model. A transition from one state to another corresponds to a choice of all enabled commands. A command is enabled in a state when its antecedent holds and all other commands with the same action synchronously hold. Depending on the type of model, the choice of transition is probabilistic, nondeterministic, or both. The PRISM tool can read and parse a model description in the PRISM language.

#### 4. Experiment Numerical Analysis

To verify the validity of the stochastic model, we have applied PRISM to implement the experiment and present the result. The experiment results of the numerical analysis had been done in this section. The performance measures considered here are several consumptions. We assume the data storage capacity of the service request is  $10(C = 10)$ , and the state transition distribution has the same value and equals 0.5. Given a performance constraint (average number of request in awaiting service), we construct the optimal policy which minimizes the average power consumption while satisfying

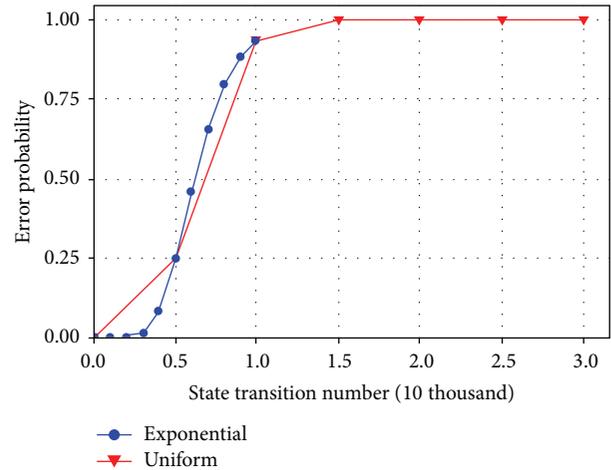


FIGURE 2: The error probability according to the changes of state transition number.

the performance constraint. We obtained these policies by constructing the generator matrix of the system in PRISM.

We start our analysis of the power error probability according to the changes of the state transition number. As shown in Figure 2, if the state transitions number increases with exponential, the power error probability of the sensor increases very quickly and may not work when the number reaches 10000. If the state transitions number increases with uniformity, the power error probability of the sensor increases very quickly, but the power error probability may keep a stable number. Based on the results of Figure 2, the number of the power error probability is large, so the state transition number under 10000 is kept.

Figure 3 shows the energy consumption when the state switches from the sleep mode to the active mode. It is clear that the sensor may consume more energy when the activities are increased. Also, from this figure, we know that the less capacity, the more the energy the sensor may consume, although the energy assumption does not change too much. Thus, from the viewpoint of minimizing the energy consumption for switching from sleep mode to active mode, it is optimal to minimize the number of data packets which may need relay through the sensor under investigation. By considering the relay is a vital property for sensor, it is imperative to find an optimal capacity to leverage the energy consumption.

#### 5. Conclusion and Future Works

As people increasingly rely on WSNs, good energy management is becoming necessary to squeeze the most out of a battery. In the paper, the states and state transitions of sensor are explicitly illustrated. According to the rules of the state transitions, a simple stochastic framework to analyze energy consumption on sensor using CMDP is proposed. We tested the framework with PRISM on both power error and power consumption according to the state transitions. We can obtain the automated analytical means by applying

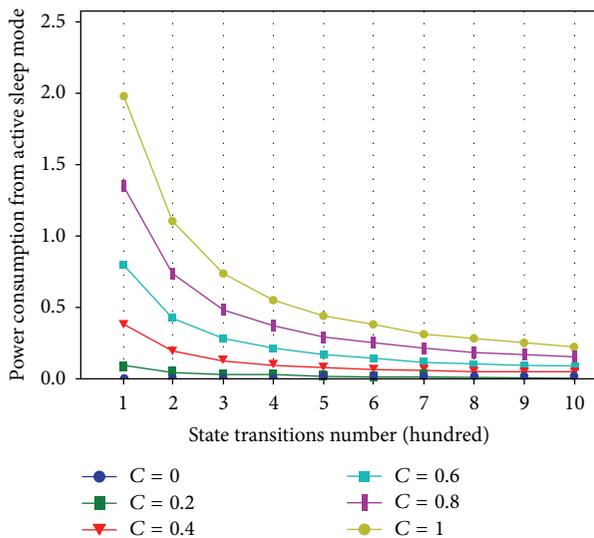


FIGURE 3: Energy consumption for switching from sleep to active state.

the given performance measures strategies with no additional expensive simulation.

In the future works, we will investigate other related results with the analytical method from this paper.

## Conflict of Interests

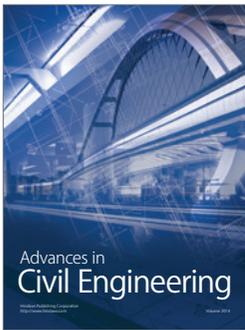
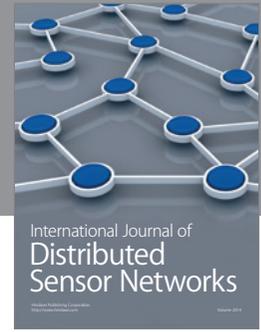
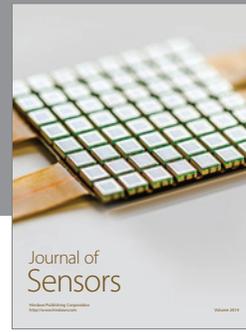
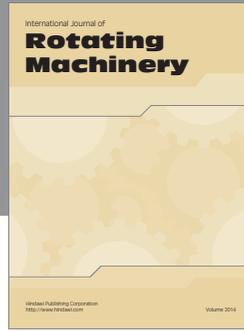
The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work was supported by “The Fundamental Research Funds for the Central Universities” (2013QNB15) and the National Science Foundation of China under Grant no. 61203304. Thanks are due to the contributors to this paper.

## References

- [1] K. Lin, K. Li, W. Xue, and Y. Bi, “A clustering hierarchy based on cellular topology for wireless sensor networks,” *International Journal of Computer Systems Science and Engineering*, vol. 24, no. 5, pp. 345–359, 2009.
- [2] Y. Yu, B. Krishnamachari, and V. K. Prasanna, “Energy-latency tradeoffs for data gathering in wireless sensor networks,” in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '04)*, pp. 244–255, Hong Kong, March 2004.
- [3] C. Zhao, M. Perillo, and W. B. Heinzelman, “General network lifetime and cost models for evaluating sensor network deployment strategies,” *IEEE Transactions on Mobile Computing*, vol. 7, no. 4, pp. 484–497, 2008.
- [4] L. A. Barroso, “The price of performance,” *ACM Queue*, vol. 3, no. 7, pp. 48–53, 2005.
- [5] A. Abdelmotalib and Z. Wu, “Power management techniques in smartphones operating systems,” *International Journal of Computer Science Issues*, vol. 9, no. 3, pp. 157–160, 2012.
- [6] I. Crk, F. Albinali, C. Gniady, and J. Hartman, “Understanding energy consumption of sensor enabled applications on mobile phones,” in *Proceedings of the 31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '09)*, pp. 6885–6888, Minneapolis, Minn, USA, 2009.
- [7] K. Lin, A. Kansal, D. Lymberopoulos, and F. Zhao, “Energy-accuracy trade-off for continuous mobile device location,” in *Proceedings of the 8th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '10)*, pp. 285–297, San Francisco, Calif, USA, June 2010.
- [8] Y. Zhang and W. Li, “An energy-based stochastic model for wireless sensor networks,” *Wireless Sensor Networks*, vol. 3, no. 9, pp. 322–328, 2011.
- [9] T. L. Cheung, K. Okamoto, F. Maker III, X. Liu, and V. Akella, “Markov decision process (MDP) framework for optimizing software on mobile phones,” in *Proceedings of the 7th ACM International Conference on Embedded Software (EMSOFT '09)*, pp. 11–20, Grenoble, France, October 2009.
- [10] P. Rong and M. Pedram, “Determining the optimal timeout values for a power-managed system based on the theory of Markovian processes: offline and online algorithms,” in *Proceedings of the Design, Automation and Test in Europe (DATE '06)*, pp. 1–6, Munich, Germany, March 2006.
- [11] K. Turitsyn, S. Backhaus, M. Ananyev, and M. Chertkov, “Smart finite state devices: a modeling framework for demand response technologies,” in *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECE '11)*, pp. 7–14, Orlando, Fla, USA, December 2011.
- [12] HTC. Htc g1 overview, [http://www.google.com/intl/en/press/pressrel/20100105\\_phone.html](http://www.google.com/intl/en/press/pressrel/20100105_phone.html).
- [13] Google. Android, official website, 2012, <http://www.android.com/>.
- [14] <http://www.prismmodelchecker.org/>.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

