Adaptive Energy Conservation Model using Dynamic Caching for Wireless Devices

Constandinos X. Mavromoustakis
Department of Informatics
Aristotle University of Thessaloniki
54124 Thessaloniki, Greece
cmavrom@csd.auth.gr

Helen D. Karatza
Department of Informatics
Aristotle University of Thessaloniki
54124 Thessaloniki, Greece
karatza@csd.auth.gr

Abstract

One important issue that has to be taken into account in wireless devices is the energy conservation. Every infrastructureless network must be adaptively self configured particularly in terms of energy, connectivity, and memory. Efficient utilization of battery power is important for wireless users because due to their movements their energy is fluctuating at different levels during operation mode. Traffic plays a major role for energy consumption because of the unpredictable incoming-flow nature [13-15]. This paper describes an adaptive method called Adaptive Dynamic Caching Energy Conservation (ADCEC), which bounds an asynchronous operation where each node evaluates dissimilar sleep-wake schedules/states based on each node's incoming sleep-history traffic. Simulation study is carried out for the energy conservation evaluation of the proposed model taking into account a number of metrics and estimation of the effects of incrementing the sleep time duration to conserve energy.

1. Introduction

Wireless devices are a collection of wireless nodes that communicate over radio. This set of mobile nodes does not need any infrastructure. These kinds of networks are very flexible and suitable for several situations and applications allowing an infrastructureless network without any preinstalled components. Every node acts as a router and has limited transmission range while the communication traffic has to be relayed over several intermediate nodes (multi-hop) to enable the communication from a source to a destination.

Nodes in wireless networks typically rely on their battery energy. In this paper, an Adaptive Energy Conservation (ADEC) protocol is examined that turns off nodes interfaces to save power of each node and maximize the network lifetime. As known in a terminal the latency, connectivity, energy and memory are the essential elements of today's mobile environments whose performance may be significantly improved by caching techniques. The ADEC protocol uses traffic patterns for the thorough examination of network behavior. Adaptively ADEC scheme assigns a variable sleeping time based on traffic that each node "accepts" throughout the sleep time duration based on a self-similarity nature of traffic. Therefore ADEC by using dynamic caching manipulate network state in a distributed form and actively control each node's sleeping duration, to minimize the energy cost of peer-to-peer communication among mobile terminals. In addition this method provides a way for guarantee the buffering of data and a way to overcome the loss of packet information. Finally this model could be applied to ad-hoc networks with any underlining routing protocol support to provide independency as well as "fair" collaboration among energy conservation mechanism and routing protocol.

The organization of the paper is as follows: Section 2 discusses the related work that has been done on energy conservation (EC) featuring out the basic energy conservation principles and conducted solutions by different schemes. Section 3 then introduces the proposed Adaptive Dynamic Caching Energy Conservation (ADCEC), followed by Section 4 which provides the evaluation and simulation results of the proposed scheme in contrast to the energy consumption associated with Adaptive Dynamic Caching. Finally, Section 5 concludes with a summary of our contribution and further research.

2. Related work

Minimizing energy consumption and maximizing the system lifetime has been one of the major design goals for wireless networks. On the one hand, many wireless device manufacturers have been striving for low power consumption in their products [2], exploring and

Proceedings of the 37th Annual Simulation Symposium (ANSS'04)
1080-241X/04 $ 20.00 © 2004 IEEE

discussing their low power transceiver architectures and low power signal processing systems. On the other hand quite a lot of protocols have been designed and use different mechanisms to reduce energy consumption, that can be classified into two categories: Active and passive protocols. Active techniques conserve energy by performing energy conscious operations, such as transmission scheduling using a directional antenna [6], and energy-aware routing [4-5]. On the other hand passive techniques conserve energy by scheduling network interface devices to the sleep mode when a node is not currently taking part in communication activity.

RADIO INTERFACE IEEE 802.11 Interfaces (2.4GHz)	TRANSMIT	RECEIVE	IDLE	SLEEP MODE	Mbps
Lucent Silver	1.3 W	0.90 W	0.74 W	.048 W	11
Lucent Bronze	1.3 W	.97W	.84 W	.066 W	2

Table 1: The degree of power consumption and Digital Radio Power States (DRPS).

Many different protocols were designed taking into account separately aspects dealing with MAC layer [7] network layer [3-5], topological and geographical information-based techniques (GAF) [8]. In [8] the entire network is divided into small virtual pieces (grids) and this area as recognizable by geographical information allows only one node to be active in the grid while the other nodes turn off their interfaces to conserve energy. In [9] the goal is to turn off nodes without significantly diminishing the capacity or connectivity of the network. The connectivity and forwarding capability as stated in [9], is maintained by keeping the nodes that constitute a backbone infrastructure (BNs [20]) in active mode, and switching off the other. However nodes' participation in network forwarding activity [1] is adaptively adjusted, based on their buttery remaining energy. In Table 1 the rates of consumption are presented for some commercial transceivers [1-5] and their Digital Radio Power States (DRPS). It is easily recognizable from different radio card samples that the rate of consumption in the "receiving" state is more than 50% of that consumed in "transmitting" state. There is importance to appropriately determine when and at what power level a mobile host should attempt transmission or retransmission of packets .

A traffic-load history determination in association with battery lifetime has not been yet investigated. In this work the overall research is focused on load history characterization for each node targeting the energy conservation concept. The self-similarity of packet traffic characterization [14, 15] enables us to examine such scenario. The proposed adaptive method allows to nodes to change their state depending entirely on their traffic history under various conditions as discussed in the following section.

3. Adaptive Dynamic Caching Energy Conservation (ADCEC)

This section describes the Adaptive Dynamic Caching Energy Conservation (ADCEC) method, which bounds a partially asynchronous operation where each node evaluates dissimilar sleep-wake schedules/states based on each node's incoming sleep-history traffic.

3.1 A scheme for dynamic caching

In a fully distributed decision system, efficient message passing plays a major role for issues concerning various QoS parameters as well as power consumption methods. As known energy conservation mechanism has to be closely collaborative with the routing protocol for the proper/acceptable behavior of nodes being in idle state and maintain the packet forwarding mechanism. But routing packets in wired/wireless communication network needs essentially a dynamic approach due to the stochastic nature of data traffic. An on demand cost-effective, adaptive solution-finding algorithm is in principle more attractive because it can keep abreast of any possible changes within network load and failures due to energy deficiency. Particularly for wireless devices/networks where nodes cooperatively form a network independently of any fixed base station infrastructure (usually infrastructureless). On the other hand in dynamic caching-oriented methods there are some trade-offs that have to be taken into account, such as generated overhead in messaging passing for delay sensitive services, network utilization, simplicity for the implementation on the wireless environment ect.

In Adaptive Dynamic Caching messages are manipulated using a specific routing protocol as will be described in the next section. Packets for routing and control purposes are contributing any time in the network for informing the neighboring nodes or all nodes in hops (route) for residual energy. Roughly speaking one of the most important task in dynamic caching is whether a node will decide to sent an acknowledgement message (ACK) to neighboring nodes or in a zone (based on the routing protocol used) for informing about the current state and the state of the next time step. If a packet needs to be sent from a source to a destination some intermediate nodes or even the destination might be in a sleep state. For intermediate nodes packets are routed via another optimal path depending on the routing protocol used. A question arises for the decision of which node will 'cache' the information for the destination node in the route (hop-by-hop basis) when the destination node or any intermediate node "lie" in the sleep mode. As known in the sleep state, node's interface can neither transmit nor receive, so it consumes energy. As a result the destined information for a proper node in sleep mode after a "timeout" will be lost.

In order to achieve an adaptive solution for this issue acknowledgement packets ACKs are sent to each neighbor in the zone (as ZRP [6, 12, 16, 17] will be explained in the next section). ACK packets are sent for informing each node about the state of the candidate node for sleep state. The mechanism used selects the previously followed node in the route (neighbor to node in the sleep mode) to cache the packets destined for the node with turned off interfaces (sleep).

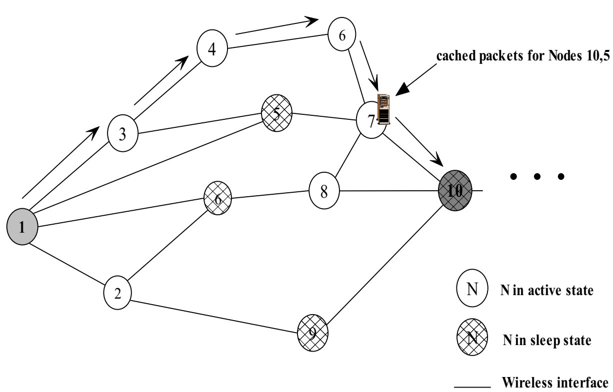

Figure 1: Wireless devices connectivity and different activity states.

This is shown in Figure 1 where packet is destined for wireless device (Node 10) from node 1. Suddenly (for triggering or energy consumption reasons) node enters into the sleep mode in order to achieve energy conservation. The time instant is not suitable since all packets destined for Node 10 will be lost (partially or in total depending on sleep time). ADCEC method enables the packets to be "cached" in the previous hop node (Node 7) from Node 10 and forwards the packets destined for Node 10 when Node 10 enters the active state. Therefore no information will be lost since it will be held in Node's 7 buffer (storage unit). Additionally the main difference from [3-9] is the adaptivity. The adaptivity occurs for the incoming traffic of the destined node [13-15]. In Figure 1 it is considered that for node N=10 the traffic is $T_{C(t),N}$ where T is the incoming traffic (capacity in the certain time t). Node 7 caches the packets for Node 10 thus increasing the storage unit for Node 10 in Node 7's unit. This corresponds to $C_{s,i}(t)$ where i is the destination node and s is the buffering node (a hop before destination). It must be also pointed out that in figure 1, node's 5 destined information could be cached also in node 3. Routing decision blocks this option since there are no memory limitations for node 7. Procedural packets PROC continuously inform for route discovery the 'route neighboring' nodes, as well as for TTL (applicable for delay sensitive services where if necessary other optimal route should be activated), residual energy, and history of all visited nodes. Furthermore PROC packets pursue one important operation: they determine whether a node can enter with safety to sleep state i.e. if there is forwarding activity in the proper time which in turn depends on the routing protocol[1], idle time and residual battery energy.

The main issue for conserving energy is to identify and associate the cached packets with the next sleeping time of the intended node. This requires the adaptivity of the cached capacity (stored packets- traffic during sleep time for Node 10) with the next-following duration of sleep time of node 10. This sleeping time will be proportionally influenced with the cached capacity in the time distance whish is $t_i(t - t_D)$, where t_D represents the equally spaced/triggered time slots. A thorough explanation of this principle is conducted in the following section.

3.2 Operations for energy conservation

When heavy load exists in wireless devices, the commonly used and known periodic sleep/listen scheme does not benefit for efficient energy conservation. The evaluation and design of energy efficient communication protocols therefore requires practical understanding of the energy consumption behavior of the underlying network interface. The main aspect for interface to consume energy is basically the mode it does operate. As known there are two basic states: The idle state and sleep state. In the idle state, an interface can transmit or receive data at any time, but it consumes more energy due to the number of circuit elements that must be powered. On the other hand in the sleep mode, an interface can neither transmit nor receive, so it consumes significantly less energy. For transmitting or receiving, an interface must explicitly transition to the idle state, which requires both time and energy. The idle energy consumption is quite high,

[1] The routing protocol used in the implementation is ZRP [12, 16].

comparable to that of receiving and in an order of magnitude more than that of sleeping [7-9], (see Table 1).

As known the periodic sleep/listen scheme does not benefit for heavy loads and conversely. In the previous section a simple scheme is described for caching the data packets destined for a proper node. The capacity of the cached data corresponds to time duration $t_i(t - t_D)$ for the next sleep duration of the destination node. Sleeping time for destination node is high enough when the node in the previous time sleeping slot, that is $t_i(t - t_{D(\tau-1)})$ did not receive any packets. Hence it stands that:

$T_{C(\tau),N} < T_{C(\tau-1),N}$ then $t_{D(\tau+1)} \geq t_{D(\tau)}$ for node N, and $\tau = 1,2,3..m$. (1)

Thus if t_D is high then the next sleep duration of node N will be in turn higher than the previous one (due to inactivity of the node in the $t_{D(\tau-1)}$). In this way each node evaluates dissimilar sleep and active states based entirely on each node's incoming sleep-history traffic (in-sleep history). This principle is shown in Figure 2.

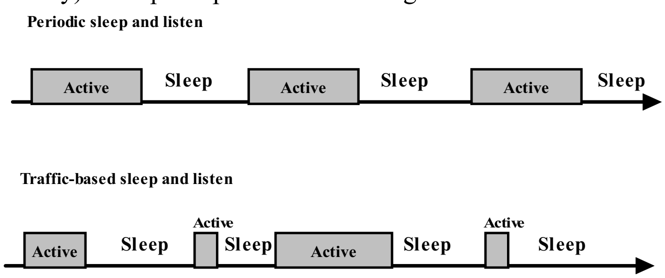

Figure 2: Different duration of sleep/wake schedules, based on node's incoming sleep-history traffic.

Sleep time duration can be measured using the following expression:

$$S(t_{new})_D = S(t)_{D(\tau-1)} - S(t)_{D(\tau-1)} \cdot \left[\frac{Packs\_cached}{Total\_packets\_destined\_for\_D} \right]$$

where $S(t_{new})_D$ is the new time duration[2] depending on the traffic history of the route destined for D, and $S(t)_{D(\tau-1)}$ is the previous time step (slot) duration.

Capacities and durations are *normalized* in order to overcome *latencies* problems. Only *latency* (usually for delay sensitive services) can "disturb" sleep period by placing a limit on the sleep time duration of the nodes. PROC packets are purely responsible for informing the node for any limitations. If a node does not receive any packets for a long period due to the stochastic nature of incoming traffic, the sleep time will increase at an unneeded grade. Thus PROC packets by using a specified field PROC_tLIM place a limit on the sleep time of the nodes to avoid increased latency thus resulting in network partitioning. Also in order to prevent a large number of nodes to enter into the sleep state and at the same time avoid network partition, we set a maximum number of nodes that are allowed to be in the sleep state (entirely based on the total number of nodes of a non-partitioned network).

For each node there are three states: the "active", the "sleep" and the "wait to sleep" or "listening" state. The state transition diagram is shown if Figure 3. Then if traffic is further reducing, after $t_{D(\tau)}$ with no incoming traffic for the time duration $t_{D(\tau-1)}$ node enters into the sleep mode. Sleep mode is only corrupted if there are serious latency limitations or sleep period expires. For these situations node "violently" enters the active state to overcome these problems.

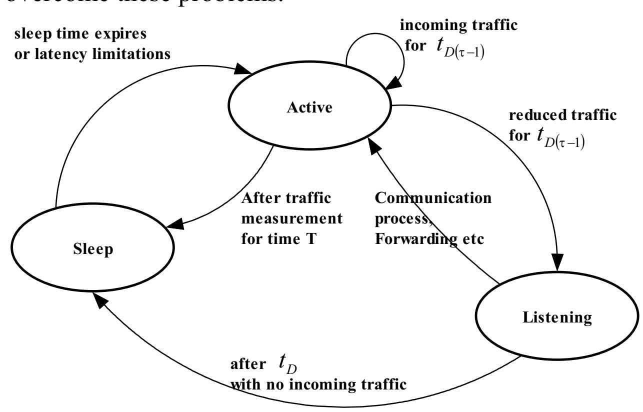

Figure 3: Node operation cycle as a state transition diagram.

Figure 3 shows the state transition diagram of each node. In the active state node's interface can transmit or receive data at any time but consuming more energy. If incoming traffic is continuously coming/destined for the node, it keeps remain in the active state. This "loop" is not energy efficient and it breaks if traffic is reduced. In this case node changes into listening mode for a while. Then if traffic is further reducing, after $t_{D(\tau)}$ with no incoming traffic for the time duration $t_{D(\tau-1)}$ node enters into the sleep mode. Sleep mode is only corrupted if there are serious latency limitations or sleep period expires. For these situations node "violently" enters the active state to overcome these problems.

Roughly speaking while traffic is not uniformly distributed, it does not allow the equally spaced sleeping

[2] New time duration cannot exceed twice the corresponding value.

Proceedings of the 37th Annual Simulation Symposium (ANSS'04)
1080-241X/04 $ 20.00 © 2004 IEEE

duration to be efficient for conserving energy. During initialization period node from active state enters the sleep state by measuring the traffic and setting a time T after activation. After network is biased [18-20] with packets, it almost rarely changes from active to sleep state by this measure. Traffic transits each node's state through the traffic history of each node during $t_{D(\tau-\Lambda)}$, where Λ is the time duration for the previous slot of that node for which data is cached.

This cooperative caching method among nodes for energy consumption bounds many aspects for mobile environments. One major aspect is the node popularity problem where multiple accesses (for forwarding purposes or being the destination) throughout time is marking the proper node resulting in node lock (which in turn results in path-route lock for time t [18, 20]). This principle is examined in the next section where an evaluation through simulation is done.

4. Simulation experiments and discussion

The design and evaluation of energy efficient communication protocols requires practical understanding of the energy consumption behavior of the underlying network interface. To demonstrate the methodology discussed in this paper, we performed exhaustive discrete time simulations of the proposed scenario under several different conditions.

4.1 Routing protocol used

One basic issue is the selection of the routing protocol that should be used in order to cooperate with the described scenario. Considering the need of bandwidth and the limited battery power for wireless devices, it is necessary to apply efficient routing algorithms to create, maintain and repair paths with least possible overhead production [2]. There are two classes of routing protocol: proactive and reactive. In proactive or table-driven protocols the routes are maintained for all possible destinations continuously-periodically, even if routes will not be actually used. The generated overhead from route maintenance cause significant reduction of network performance, increase in end-to-end delays and delay variations. Reactive or on-demand protocols on the other hand, create and maintain routes only when they are needed.

In the implementation of the proposed scenario the Zone Routing Protocol (ZRP) [6, 12, 16, and 17] is used. ZRP is a hybrid protocol that combines the reactive and proactive modes. The ZRP is considered advantageous because allows to a certain node to accurately know the neighbors of any mobile terminal within a zone. These devices should be in zone that could be accessible in a fixed number of hops. Since ZRP allow the absolute communication with neighbors, is considered less expensive, while neighbors contribute in the routing process [12]. Particularly ZRP divides the network into several routing zones specifying a determined number of hops. This allows the routing protocol to be adjustable for different operational network conditions such as heavy traffic [13-15].

4.2 Simulation results of the proposed scenario

To emulate the scenario described earlier, the need of a possible realistic environment must be achieved. In this section, we present some experimental and simulation results for performance evaluation and energy conservation offered by our scheme. The power could be measured by monitoring the three basic metrics-energy components: (i) *transmission* power required to send a packet, (ii) *reception* power required to receive or listen to a packet, and (iii) *idle* power required to stay in at the active state (awake) in contrast with sleep time duration that follows. Transmission power includes both the power required to drive the circuit and the transmission energy from the antenna [1, 2]. Therefore, the energy consumed by any mobile terminal for sending, receiving or discarding a message is given by the linear equation[3] [1] $Energy = m \cdot size + \beta$; where *size* is the message size, and m denotes the incremental energy cost associated with the message and β a fixed cost of each operation.

The energy consumption model used in the simulation, for the calculation of the amount of energy consumed, is based theoretically on the WaveLAN PC/Card energy consumption characteristics found in study by Feeney and Nilsson [1].

Two sets of experiments were performed. One set deals with the caching concept and the grade of contribution in conserving energy, and the second deals with the energy conserved under significant traffic and network partition limitations and the latency issues that arise. As mentioned the caching capacity of each node could be unlimited while nowadays memory becomes cheaper and cheaper. An issue then arises from node's unlimited capacity because if the traffic will reach a high capacity destined for a node at the sleep state, then latency and delay problems will occur while the forwarding mechanism takes place. Thus taking into

[3] Linear regression is used to test the model and find values for m and β.

account this issue, in this paper two different types of caching capacity have been evaluated. Investigation has been performed for the following: (i) Unlimited capacity (ii) Limited capacity for each node as 64KB, 128 KB, 256 KB, 512 KB, 1 MB, 2MB.

An issue that has to be taken into account is whether the cached information destined for a proper node could be stored in a node with higher residual energy. As shown in simulation process if nodes with higher level of residual energy are chosen in the path then the network partitioning probability is further reduced. For this reason cached information is chosen on a recursive path basis, where source node while having the path tries to find the node with the higher residual energy in order to assign the caching process.

Another issue that arises for wireless devices is the dynamically changing topology of the network. If a caching information process takes place and a node or some nodes in the path will change their state into sleep mode then the path simultaneously changes having at least the 'caching' node within the new path. If this enterprise is impossible due to sudden network partitioning then before network splits, the node which caches the information forwards the cached information to destination which violently enters the active state.

In simulation was used a two-dimensional network, consisting of 25 nodes with each link (frequency channel) having max speed reaching 2Mb per sec. The propagation path loss is the two-ray model without fading. The network traffic is modeled by generating constant bit rate (CBR) flows. Packets generated at every time step by following Pareto distribution (2) as depicted in [13-15], destined for a random destination uniformly selected.

$$P(X = x) = \frac{a \cdot b^a}{x^{a+1}}, \text{ with } E(x) = \frac{a \cdot b}{a-1}. \quad (2)$$

Equation (2) generates the probability density function of the Pareto distribution and the corresponding mean value. Where a is a shape parameter and b is the minimum value of x. When $a \leq 2$ the variance of the distribution is infinite. For the generation of self-similar traffic [14], a should be between the values of 1 and 2. Roughly speaking the load generated by one source is mean size of a packet train divided over mean size of packet train and mean size of inter-train gap or it is the mean size of ON period over mean size of ON and OFF periods as follows (3):

$$L_i = \frac{\overline{ON_i}}{\overline{ON_i} + \overline{OFF_i}} \quad (3)$$

Additionally we have modeled in each node an agent which evaluates the information destined for a proper destination. In this way we have at any time measures of the information destined for each node (for a given time interval) by any node. This has been implemented as a [N-1] row, [N-1] column for each node being a possible destination.

Figure 4 illustrates the periodic sleep time in comparison with the non-periodic sleep time. As seen the periodic sleep time is almost the half time of that of the device "live". In contrary the non-periodic time, where each node evaluates dissimilar sleep and active states based on each node's incoming sleep-history traffic, shows a significant reduction in active period and an increase in the mean sleep time. During simulation time it was shown that sleep time should not increase more than 71% at a time from previous measure (as peak, and not simultaneously but in different slots) because this will cause network partitioning and routing lock [18]. According to fig. 4 sleep time increases by almost 10% mean (non-periodic time) in comparison with the sleep time occurring on a periodic basis.

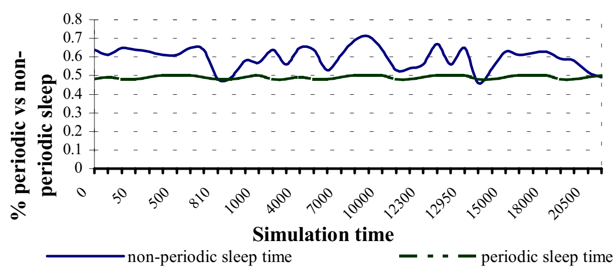

Figure 4: Comparison of periodic sleep time with non-periodic (unlimited capacity).

Figure 5 shows the average hop number at any time during simulation, with the cached information. It is clearly shown that for small number of hops the cached information is negligible (in terms of few Kbs). On the other hand for relatively higher number of hops the average cached information in KB are increasing, but not exceeding the 3.87 MB (extremely high traffic-network splits). Of course these measures have been taken for unlimited capacity used for caching in each node's buffer.

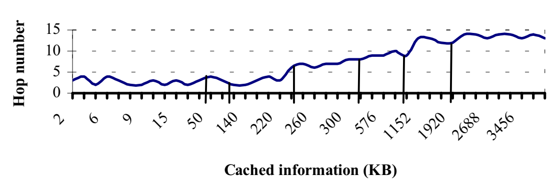

Figure 5: Illustration of the average hop number with respect to cached information.

Figure 6 clearly shows the frequency of caching technique necessary for saving packets destined for a proper node, which prevents the loss of packets. Having 0.3 as a mean value of the sleep time, the node which caches packets destined for a node laying in the sleep state, saves the packets by using the caching technique.

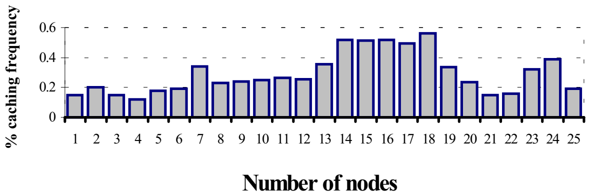

Figure 6: How frequently each node caches information for any other node (depending on network topology and node's mobility-in the presence of excessive traffic).

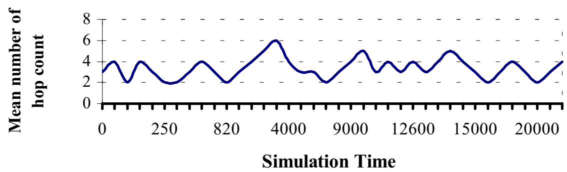

Figure 7: Mean number of hop count at any time in the network.

Figures 7 and 8 show the mean number of hop count at any time in the network and the mean number of nodes 'laying' in the sleep state respectively. It is remarkable to point out that nodes in order to prevent network partitioning in the sleep state must not exceed the limit of 10.

From fig. 9 it is indicated that unlimited capacity in each device could enable vulnerabilities for network connectivity. Roughly speaking when memory is unlimited, after some simulation steps (by reaching steady state [18-19]) nodes will enter in a higher duration sleep time period. As a result network could split in parts (partitioning) causing significant reduction in performance, even if a limited number of nodes are allowed to enter in the sleep state. Hence a limited capacity for caching information destined for other nodes as shown in fig. 9 might offer better connectivity and in turn network partitioning prevention.

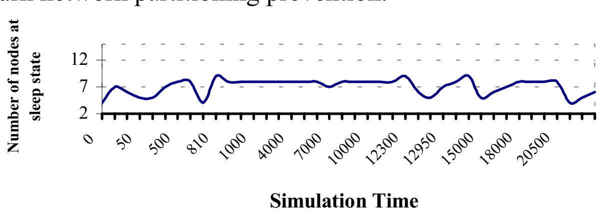

Figure 8: Mean number of nodes 'laying' in the sleep state (excessive traffic).

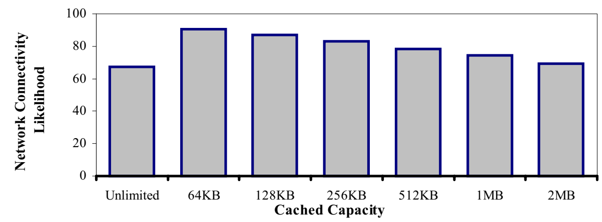

Figure 9: Network connectivity maintenance with respect to different capacity measures.

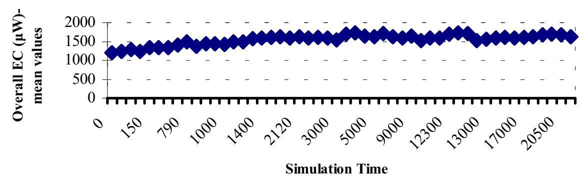

Figure 10: Mean Energy consumption in the network at any time during simulation.

For 64, 128, 256, 512 KB cached capacity it can be seen that for non-periodic sleep time duration ADCEC offers higher connectivity even in the presence of extremely high traffic. Thus in the implementation of the described scenario when latency for transmitting the corresponding packets to destination increases, then a limitation in sleep time duration must be placed for all nodes. Figure 9 basically ensures the fact that unlimited memory for caching is not useful particularly for dynamically changing network. Thus placing a limit on caching capacity would enable higher network connectivity and better battery utilization by increasing the average sleep time duration. Finally in fig. 10 the mean energy consumption at any time during simulation in the network is presented.

5. Conclusions and further research

In this paper, we have implemented an Adaptive Dynamic Caching Energy Conservation (ADCEC) method, which bounds a partially asynchronous operation where each node evaluates dissimilar sleep-wake schedules based on each node's incoming sleep-history traffic. One basic issue outlined for conserving energy was the association of cached packets with the next sleep time duration of the intended node. Results have shown that unlimited memory for caching in wireless networks isn't useful compared with certain caching capacities and specified mobility frequency. Additionally it was shown that ADCEC method would enable a significant increase in the total average sleep time duration, the same time keeping network connectivity at high levels (reliability) which results in remarkable energy conservation.

If a wireless device needs to be connected with internet via other wireless networks or directly, a scope of interest could be the capabilities of such an enterprise to service mobile users on demand. The main challenge in wireless multi-hop ad-hoc networks is the efficient routing problem, which is aggravated by the node mobility. Therefore a routing technique that would enable an efficient routing scheme for web information retrieval by wireless devices for infrastructure-based mode or even by using the ad-hoc-based mode becomes necessity with the tremendous growth of mobile users intending to access the web. Furthermore a scope of interest would be the exploration of an agent-based approach [18] for web-caching mechanism where an asynchronous sleep state will also exists (using epochs [21] for activation period).

References

[1] L. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In proc. of IEEE InfoCom, 5(8), 2001.

[2] Porret, T. Melly, C. C. Enz, and E. A.Vittoz. A Low-Power Low-Voltage Transceiver Architecture Suitable for Wireless Distributed Sensors Network. In proceeding of IEEE International Symposium on Circuits and Systems'00 Geneva, 2000; Volume 1: 56-59.

[3] Ben Liang, Zygmunt J. Haas. Optimizing Route-Cache Lifetime in Ad Hoc Networks. INFOCOM 2003.

[4] S. Banerjee and A. Misra. Minimum Energy Paths for Reliable Communication in Multi-hop Wireless Networks, in The ACM Symposium on Mobile Adhoc Networking and Computing (MOBIHOC 2002), Lausanne, Switzerland, June 9-11 2002.

[5] S. Doshi and T. X. Brown. Minimum Energy Routing Schemes for a Wireless Ad Hoc Network, IEEE InfoCom 2002, New York, NY, June 23-27 2002.

[6] Z. J. Haas and M. Pearlman. The performance of query control schemes for the zone routing protocol. ACM/IEEE Transactions on Networking 9(4), 2001.

[7] S. Singh and C. Raghavendra. PAMAS: Power Aware Multi-Access Protocol with Signalling for Ad Hoc Networks, SIGCOMM Computer Communication Review, vol. 28, no. 3, pp.5-26, July 1998.

[8] Y. Xu, J. Heidemann, and D. Estrin. Geography-informed Energy Conservation for Ad Hoc Routing, MobiCom'2001, pp. 70-84, Rome, Italy, July 2001.

[9] B. Chen, K. Jamieson, H. Balakrishnan and R. Morris. Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks, in MobiCom'2001, pp. 85-96, Rome, Italy, July 2001.

[10] Joanna Kulik et al. Negotiation-based Protocols for Disseminating Information in Wireless Sensor Networks, Wireless Networks, vol. 8, no. 2-3, pp.169-185, March-May 2002.

[11] S. McCanne and S. Floyd. NS network simulator version 2.1b8, June 2001, http://www-mash.cs.berkeley.edu/ns/ .

[12] Z. J. Haas. A new routing protocol for the reconfigurable wireless networks. In Proc. of the IEEE ICUP'97, 1997.

[13] K. Park, G. Kim, and M. Crovella. On the relationship between file sizes, transport protocols and self-similar network traffic. In ICNP '96, 1996.

[14] S. Robert and J.-Y. Le Boudec. New models for pseudo self-similar traffic. Performance Evaluation, 30:57--68, 1997.

[15] P. Ulanovs, E. Petersons. Modelling methods of self-similar traffic for network performance evaluation, Scientific Proceedings of RTU. Series 7. Telecommunications and Electronics, 2002.

[16] Z.J. Haas and M.R. Pearlman. The Zone Routing Protocol (ZRP) for Ad Hoc Networks, Internet Draft, draft-ietf-manet-zone-zrp-02.txt, June 1999.

[17] M. R. Pearlman, J. Deng, B. Liang, and Z. J. Haas. Elective Participation in Ad Hoc Networks Based on Energy Consumption, in IEEE GLOBECOM 2002, Taipei, Taiwan, November 17-21 2002.

[18] C. X. Mavromoustakis, H. D. Karatza. Agent-based throughput response in presence of node and/or link failure (on demand) for circuit switched telecommunication networks, to appear in Computer Communications, Elsevier (accepted for publication).

[19] C. X. Mavromoustakis, H. D. Karatza. On the extensibility properties and performance measures of circuit switched telecommunication networks, using agent-based distributed routing algorithm, Proceedings of 2003 International Symposium on Performance Evaluation of Computer and Telecommunication Systems-SPECTS, Montreal, Canada, July 20 - 24, 2003, pp. 240-247.

[20] C. X. Mavromoustakis, H. D. Karatza. Split agent-based routing in interconnected networks. To appear in International Journal of Communication Systems (IJCS), Wiley (accepted for publication).

[21] H. D. Karatza and R.H Hilzer. A Simulation-Based Performance Analysis of Epoch Load Sharing in Distributed Systems, Simulation: Transactions of the Society for Modelling and Simulation International, 2002, Vol. 78(7), pp. 461-471.