Energy and Time Trade-Offs in Duplicate Packet Transmission

E. Gelenbe
Dept. of Electrical & Electronic Engineering
Imperial College
London SW7 2BT, UK
e.gelenbe@imperial.ac.uk

A. Filippoupolitis
Dept. of Electrical & Electronic Engineering
Imperial College
London SW7 2BT, UK
afil@imperial.ac.uk

I. Eid
Dept. of Electrical & Electronic Engineering
Imperial College
London SW7 2BT, UK
imad.eid@imperial.ac.uk

Keywords: Sensor Networks, Autonomic Communication, Packet Travel Time

Abstract

Sensor and ad hoc networks are designed to function in uncertain environments. The transmission medium, the possible mobility and unreliability of nodes, and possible malicious interventions by adversaries, create significant uncertainty and increase the effective transmission delays, which in turn may be mitigated by appropriate protocols. One obvious approach to mitigating uncertainty is to forward information over multiple independent paths, but this means of reducing overall unreliability comes at the cost of increased energy consumption as well as the creating of congestion and possible conflicts and packet collisions. In recent papers [1] theoretical models have been developed for packet travel times in a wireless network whose nodes are distributed over a volume of space, and in which packets have to travel over a random number of multiple relay hops before they reach their destination. In this paper we present detailed simulation studies in a variety of wireless topologies to investigate the trade-off between reliable packet delivery, the effective packet delivery time, and the energy cost per packet in wireless networks. Our simulations include unfavourable conditions, such as when the packet is provided with wrong information about the direction that it should follow, and include the effect of time-outs and packet retransmission when it is assumed that the packet has been lost. We observe that under specific conditions, the simultaneous transmission of duplicate packets can lead to improved overall performance, in significantly reducing travel effective delay without significantly increasing energy expenditure.

1. INTRODUCTION

The travel time of a packet in a wireless sensor network is affected by numerous factors [2, 3, 4]. The mobility of the nodes, the existence of possible physical obstacles or of uncertainties in the node's location, the characteristics of the physical medium, as well as the algorithms that forward packets, all play an important role with respect to the time a packet needs to reach the destination. The uncertainty of these transmission media and the complexity of the physical environment has made necessary the development of novel and reliable packet forwarding techniques.

Consider a wireless network where the nodes are distributed over an area or volume of space so that when a packet is transmitted it may needs to travel over a random number of hops until it reaches its destination. We consider a source node A that wishes to send a packet to a fixed destination node B. All the nodes in the network have a wireless range d. The physical distance between the two nodes is J. We make the assumption that $J >> d$ in order to ensure that multiple hops are needed until the packet reaches node B. Instead of using the physical distance J, we can define a distance $D = \frac{J}{d}$ which is expressed in terms of the minimum number of hops that are necessary in order to reach node B. Furthermore, we assume that some intermediate nodes may fail or may be out of range of each other. We then have to address the problem of estimating the time T_1 it takes a packet to travel from node A to node B given that the initial distance between the two nodes is D. This question is relevant both to the evaluation of wireless sensor networks [5, 6, 2] and to ad hoc network performance [7, 8, 3], and has been considered by several authors [9, 10, 11]. It is a mathematically interesting and very difficult problem even if simplifying assumptions are made about the location of the nodes [11, 10]. It has not been solved yet in an exact manner as the available results only offer bounds and approximations for the packet travel time. Finally, it can also be of practical interest to wired networks as well, which use local search techniques for packet routing [12, 13, 14] instead of deterministic shortest-path techniques.

In this paper we present a simulation study to investigate the packet travel time and the respective energy expenditure in a wireless sensor network. We first recall some results of a mathematical model [1] based on diffusion approximations that estimates the amount of time it will take a packet to go from its source to the destination. In Section 2 we describe the simulator tool and illustrate the simulation results for different conditions and scenarios. We also study the effect of simultaneous packet transmission on the packet travel time and on the energy expenditure.

1.1. A Model For Packet Travel Time

The amount of time it takes a packet to reach its destination is influenced by numerous factors that introduce randomness. For instance, it is possible that the nodes are not placed at regular intervals. The signal propagation may be obstructed by physical objects. Furthermore, a node can choose a neighbour to forward its packet based on various criteria such as the quality of the signal or his estimate of the "best" neighbour. All these factors introduce randomness to the amount of time that a packet will need in order to reach the destination.

We assume that the instantaneous distance of the packet to its destination is Y_t. When the packet has reached its destination at time T_1, the distance $Y_t = 0$, and its travel will stop. While the packet is travelling towards the destination, so that $t \leq T_1$, it may be lost. Considering a small time interval $[t, t + \Delta t]$, we can express the probability of loss by $\lambda\Delta t + o(\Delta t)$. This event can occur due to lack of active nodes inside the transmission range or because of some transmission error. Consequently, the probability that the packet is received by a relay node is $1-\lambda\Delta t+o(\Delta t)$. We represent by $b\Delta t$ the mean change over the time $[t, t+\Delta t]$ of the distance of the packet to the destination. We use $c\Delta t$ to represent the variance of the distance that the packet has travelled in the same time interval. Both b and c are constants and do not depend neither on the distance to the destination nor on the time. This implies that the nodes' distribution is homogeneous and that the system characteristics remain unchanged over time. By transforming the transient process of the packet's travel to an ergodic one and following the analysis in [1], we have the following result: In the case where there is no packet loss, no time-out and the packet makes, on average, a progress towards the destination($b < 0$), then the mean time it takes the packet to reach its destination is

$$E[T] = \frac{D}{-b} \qquad (1)$$

This formula guarantees the feasibility of the search. In other words, if $b < 0$ then the average search time will be finite. The value of c, which is the second moment of the distance traversed by the packet per unit time, will not affect the average travel time.

If the source node has not received an acknowledgement from the receiver by a certain time, it will incorporate a time-out mechanism in order to retransmit the packet. Furthermore, a packet that has not successfully reached its destination by a certain number of hops may destroy itself. We make the assumption that the packet uses a time-out of length τ. When this time-out expires the packet self-destroys. We assume that the time-out is an exponentially distributed random variable with parameter $r \geq 0$. Its average value is $E[\tau] = \frac{1}{r}$. The source will retransmit the packet after the expiration of the time-out. But in order to avoid retransmitting packets that may have reached their destination, we introduce a delay M which elapses before the source sends a duplicate packet. We again assume that M is an exponentially distributed random variable with parameter μ and its average value is $E[M] = \frac{1}{\mu}$. The usage of the time M guarantees that the source will not retransmit a packet too soon after the time-out is invoked. Thus we avoid retransmission of packets that successfully reached the destination without the source having been informed of the success. We are going to calculate the total time it takes a packet to reach its destination after multiple retransmissions. The result we are going to illustrate includes the impact of possible packet losses (λ), of the time-out delay (τ) and of the overhead delay (M). We are going to assume that a time-out or a loss of a packet may occur at any distance from the destination, except of course when we have reached the destination. We must also note that while the packet is travelling in the network it may reach a position where its distance to the destination is equal or even larger than its initial value. A detailed analysis for the mean packet travel time $E[T]$ can be found in [1].The result we have is that the total average time which is possible to include several restarts after time-outs and retransmission overhead delays, or losses with time-outs and overhead delays, is:

$$E[T] = -2D\frac{1+\frac{\lambda+r}{\mu}+\frac{\lambda}{r}}{b-\sqrt{b^2+2c(\lambda+r)}} \qquad (2)$$

The favourable situation occurs when $b \leq 0$, so that the packet makes some progress towards its destination each time it is forwarded. We can verify that in the case where there is neither a time-out ($r = 0$) nor packet loss ($\lambda = 0$) Equation 2 reverts to Equation 1. We can rewrite Equation 2 in the following form:

$$E[T] = 2D\frac{1+\frac{\lambda+r}{\mu}+\frac{\lambda}{r}}{|b|+\sqrt{b^2+2c(\lambda+r)}} \qquad (3)$$

We can verify that as $r \to \infty$, $E[T] \to \infty$. This refers to the case where the timeout value is very small and the packet is destroyed before it manages to reach the destination. When $r \to 0$ we also have $E[T] \to \infty$. This is the case where the timeout value is large and packets are drifting away from the destination and are not destroyed. Given that $E[T] > 0$ and it is continuous with respect to r, it will have a minimum value.

In Figure 1 we can see that the average time $E[T]$ is influenced by the timeout and by the variance parameter c. When packet loss can occur, an increased variance parameter c reduces the average search time. This is because a larger variance value increases the possibility of finding a shorter path, since losses are followed by restarts. We must also note that there is an optimum value for the timeout which minimises the average time.

A very interesting and surprising result occurs when $b = 0$. This corresponds to a case where the packet is forwarded

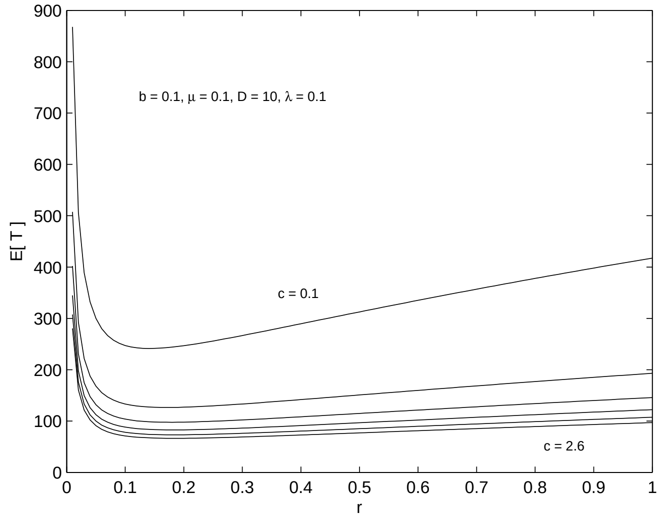

Figure 1. Mean packet travel time versus time-out rate, for D=10, average value of the overhead delay M=10, loss rate $\lambda = 0.1$ and different values of the variance parameter c=0.1, 0.6, 1.1, 1.6, 2.1, 2.6.

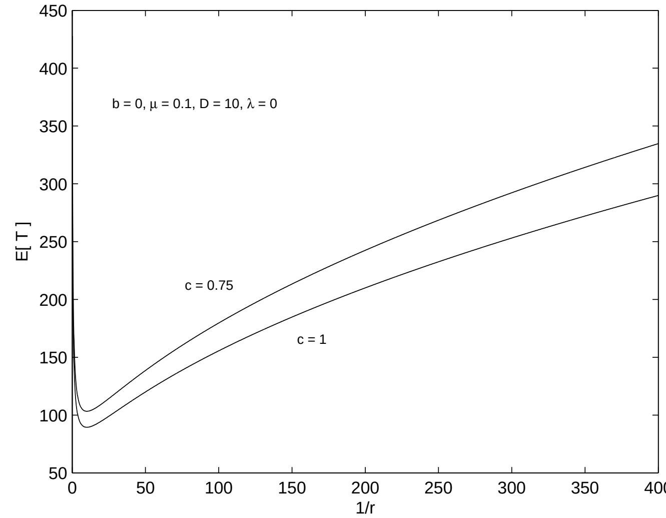

Figure 2. Mean packet travel time versus the average value of the exponentially distributed timeout $E[\tau] = \frac{1}{r}$. We have no losses, source to destination distance $D = 10$, an exponential value of overhead delay before packet retransmission M=10 and a variance parameter c=1 for wireless range d=1, and c=0.75 for $d = \sqrt{2}$

under perfect ignorance. In other words, at each step it neither gets further away nor closer to the destination. If we can have packet loss ($\lambda > 0$) or if there is a time-out ($r \geq 0$), the travel time of the packet is given by:

$$E[T] = 2D\frac{1 + \frac{\lambda+r}{\mu} + \frac{\lambda}{r}}{\sqrt{2c(\lambda + r)}} \quad (4)$$

So even when we search for the destination without a clear direction which way to go, the packet will make it in a finite amount of time on the average. Figure 2 illustrates there is an optimum value for the time-out $\frac{1}{r}$ which results in the smallest value of $E[T]$.

2. A SQUARE GRID TOPOLOGY SIMULATION

We have developed a simulator tool in order to verify the preceding theoretical analysis. Each node in the network has a fixed wireless range d. This means that a packet has to traverse multiple hops in order to go from the source to the destination. We also assume that the nodes are reliable. They will successfully receive a packet and will also retransmit it if this is necessary. Another assumption we make is that there is no queueing. When a packet arrives at a node, it will not have to wait in order to be relayed. The duration of a packet transmission between two nodes is defined as a unit hop time. There is no packet loss in our simulation , due to the nodes being reliable. Furthermore, there are no collisions between packet transmissions at any of the nodes.

An important parameter of the simulation is the *time-out*. It is defined as the maximum number of hops that a packet can traverse before it is destroyed. When the packet has travelled for a number of hops greater than this threshold, it is discarded and the source begins its retransmission. This mechanism makes sure that the packet will finally reach its destination when it has no knowledge or imperfect knowledge of the direction it has to take. Before the retransmission of a discarded packet, we wait for a number of hops equal to the overhead delay. The timeout value and the overhead delay value have been considered constant in our simulation.

The topology of the network is a square grid. The study of other topologies, where nodes are distributed according to a proability distribution, could be a part of our future work. The wireless nodes are placed on an infinite plane at regular unit intervals. A node in the grid will be the source that transmits the packets to a fixed destination node. We have assumed that a node has no knowledge of the right direction towards the destination. Thus, when forwarding a packet it will randomly select one of its neighbour nodes. Depending on the transmission range of each node, we have considered the following two cases.

In the first case we assume that the transmission range of each node is $d = 1$. This means that a node with coordinates (i, j) can have exactly four neighbours: $(i \pm 1, j)$ and $(i, j \pm 1)$. There is an equal probability of 0.25 that one the node's neighbours will be the destination of the packet. In the second case we set the value of $d = \sqrt{2}$. Now a node (i, j) can communicate with eight neighbours: $(i \pm 1, j)$, $(i, j \pm 1)$, $(i+1, j+1)$, $(i-1, j-1)$, $(i+1, j-1)$ and $(i-1, j+1)$. The probability that a packet is forwarded to a neighbour node is 0.125.

The distance D between a source node (i, j) and a destination node (l, k) is: $D = |i - l| + |j - k|$. Let us denote by δ the current distance between the source and the destination. We have to consider the following cases:

- If $\delta > 1$ and $l \neq i$ and $k \neq j$ then the new distance will be either $\delta - 1$ or $\delta + 1$, each with probability $\frac{1}{2}$. We will call this case "unaligned"

- If either $l = i$ or $k = j$, then with probability $\frac{3}{4}$ the new distance will be $\delta + 1$ and with probability $\frac{1}{4}$ it will be $\delta - 1$. We will call this case "aligned".

Considering the case where the wireless range $d = 1$, there is an equal probability of 0.25 that one the node's neighbours will receive the transmitted packet. In the aligned case the value of $b = +0.5$ whereas in the unaligned case the value of $b = 0$. In both cases $c = 1$. Thus we notice that the value of b depends on the relative positions of the intermediate and the destination node. More generally, for any value of δ there will be $(\delta + 1 - 2) \cdot 4$ unaligned nodes and 4 aligned nodes. So we can see that the aligned case becomes more frequent as we get closer to the destination node. In the case where $d = \sqrt{2}$, the average drift will be zero for both aligned and unaligned nodes. If the distance to the destination is 1 than $b = +0.25$. Since for any node there are two neighbours (left and right node) for which the change in the distance to the destination is zero, the value of $c = 0.75$.

2.1. Simulation Results for Packet Travel Time

Using the simulator, we investigated the case where the forwarding of the packet was done under perfect ignorance. This means that at each step the packet is forwarded to a *random* neighbour of the current node. The initial distance D between the source and the destination node is 10 hops. For each (constant) value of the timeout, we conducted 500 simulation runs and calculated the mean value of the hops that where needed in order for the packet to reach the destination node. Figure 3 illustrates the results for the case where each node has a wireless range $d = 1$, while Figure 4 depicts the case where $d = \sqrt{2}$. The value of the overhead delay in both cases is $M = 10$ hops.

As we can note from Figures 3 and 4, the packet reaches its destination although it is being forwarded under perfect ignorance. It is also evident that the number of hops is strongly

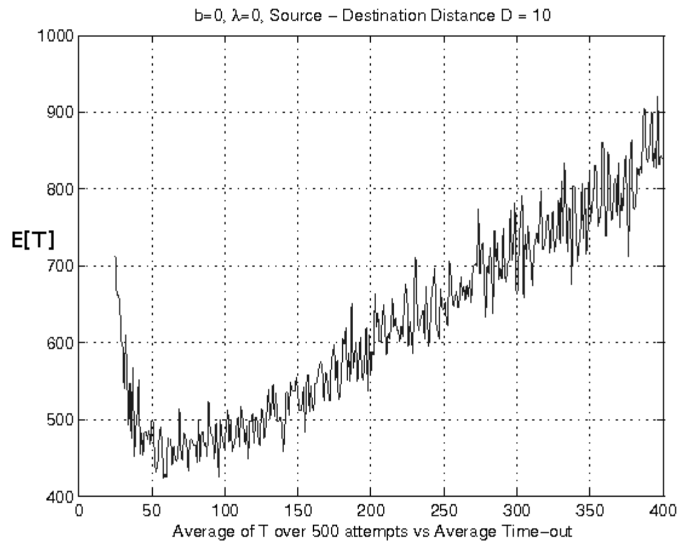

Figure 3. Mean number of hops for the 4 neighbours case (Wireless range $d = 1$). The source to destination distance $D = 10$, the value of $b = 0$ and the value of $c = 1$. The wait time before packet retransmission is $M = 10$.

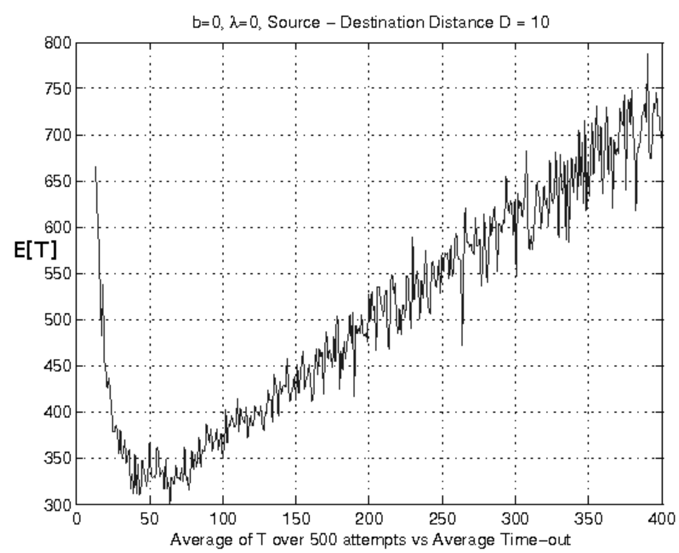

Figure 4. Mean number of hops for the 8 neighbours case (Wireless range $d = \sqrt{2}$). The source to destination distance $D = 10$, the value of $b = 0$ and the value of $c = 0.75$. The wait time before packet retransmission is $M = 10$.

affected by the value of the timeout. For small values of the timeout, the packet has to travel a great number of hops before it reaches its destination. However, we can clearly note that there is an optimal value for the timeout where the number of hops is minimum. Beyond this timeout value, the number of hops begins to increase. Comparing these results with the theoretical ones of the previous section, we can confirm that the curve has the same general form. We must also note that the optimum value of the time-out is in comparable ranges in both cases. The theoretical results are, however, optimistic compared to the simulation results. A significant reason for this difference is the fact that the packet tends to drift away as it approaches the destination, but the theoretical model that was used does not include a dependence of b on the distance.

2.2. Simultaneous Packet Transmission

In this section we study the simultaneous transmission of several copies of a packet in the network. Since the copies are forwarded randomly, they will follow different routes. Thus we expect that it is more likely for one of the multiple packets to reach the destination within a shorter time, compared with the case where only one packet was transmitted. Once the first copy reaches its destination, its travel time is the packet travel time. We ran a simulation with a fixed value for the time-out parameter (100 hops). The parameter that is going to vary is the number of simultaneous transmissions for a packet. For each value of the packet transmissions number, we keep the travel time that was achieved. We investigated two different topology scenarios depending on the number of neighbours a node can communicate with (four or eight). The results are depicted in Figure 5. Each point in the graph is the average of 1000 simulation runs.

The simulations were conducted with a fixed value of 100 hops for the timeout. The average travel time of a packet towards its destination depends on the timeout value. Since we used a constant timeout value of 100 hops, the average travel time of the packets must be the same although we have different number of transmissions. We calculated the average travel time of the packets for both topology scenarios. For the case of 4 neighbours, the average travel time of a packet is 479 hops. For the case of 8 neighbours the average travel time is 372 hops. These values agree with the simulation results of the Average Travel time vs. the Average Timeout value as we can confirm by Figures 3 and 4. We must also note that the value of the Average Travel time affects the Minimum Travel time of the packet. This can be seen in Figure 6. We have used three different values for the timeout: 100 hops, 200 hops and 400 hops. This gave us three different values for the Average Travel time as we can see in Table 1. These values also agree with the graph in Figure 3.

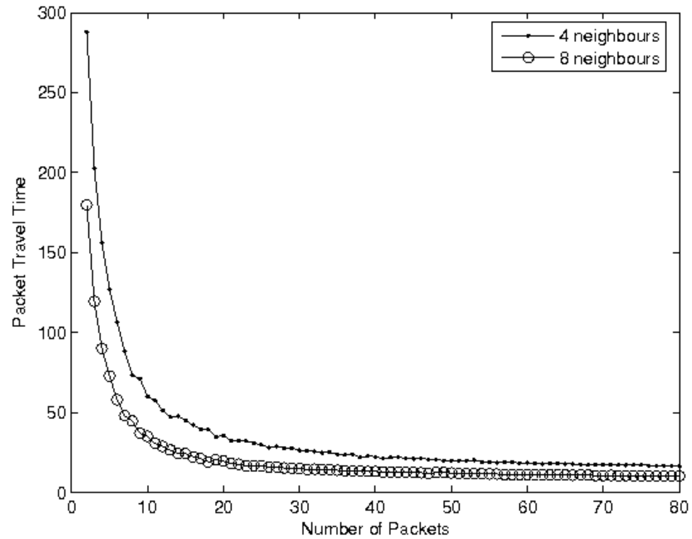

Figure 5. Average Packet travel time vs. Number of Simultaneously Transmitted Packets. We illustrate the cases where the wireless range $d = 1$ and $d = \sqrt{2}$. The source to destination distance $D = 10$, and the wait time before packet retransmission is $M = 10$.

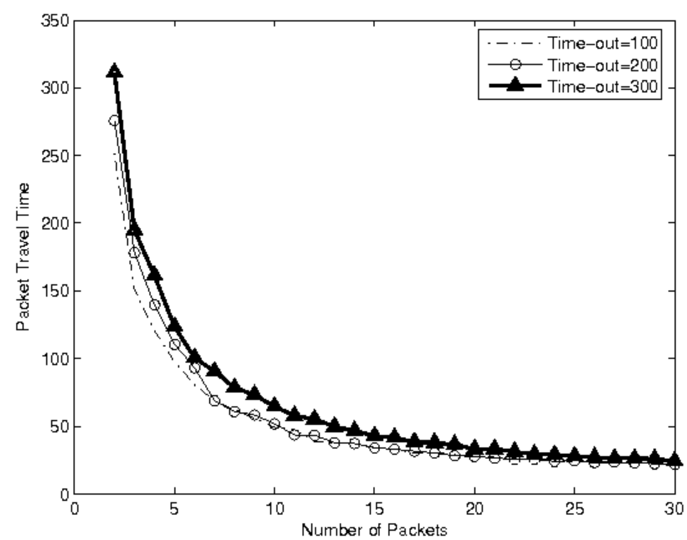

Figure 6. Comparison of average packet travel time for simultaneous packet transmissions and different timeout values. Each node has 4 neighbours (Wireless range $d = 1$). The source to destination distance $D = 10$, the value of $b = 0$ and the value of $c = 1$. The wait time before packet retransmission is $M = 10$.

Timeout(hops)	Average Travel Time(hops)
100	372
200	495
300	720

Table 1. Average Travel Times for different Timeout values (8 neighbours case).

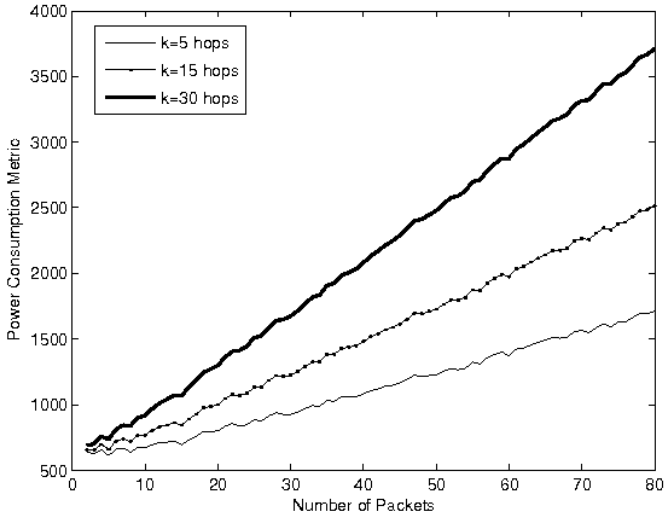

Figure 7. Power Consumption Metric vs. Number of simultaneously Transmitted packets. Each node has 4 neighbours (Wireless range $d = \sqrt{2}$). The source to destination distance $D = 10$, the value of $b = 0$ and the value of $c = 1$. The wait time before packet retransmission is $M = 10$.

2.3. Energy Expenditure

Based on the average travel time that a packet needs in order to reach its destination, we can define a power consumption metric. Let us denote by T_m the average packet travel time when we simultaneously transmit multiple packets in the network. Let the number of simultaneously transmitted packet be N. However, we must note that when the first copy of a packet reaches the destination, the rest of the copies will keep being transmitted in the network until the nodes are informed that there was a successfull transmission or until they are self-destroyed. Thus, we must introduce an overhead k that will express the number of hops that will elapse before the rest of the simultaneously transmitted packets stop propagating. Now we can define the power consumption metric P as:

$$P = (T_m + k) \times N \tag{5}$$

In order to minimise P we must minimise the Packet Travel Time (T_m) or the Number of Simultaneously Transmitted packets (N). As Figure 5 shows, minimising the Number of Transmitted Packets will result in an increased Average Packet Travel time. So there is a tradeoff between the number of packets we transmit and the Average Packet Travel time value. However, we can minimise the Average Travel Time by using the optimal value for the timeout, which is clearly shown in Figures 4 and 3. In order to better understand the behaviour of the network in terms of the power consumption metric we repeat the simulation. Figure 7 shows the power consumption metric P versus the number of simultaneously transmitted packets.

As we can see in the previous Figure, the number of simultaneously transmitted packets affects the value of the power consumption metric. As we increase the number of copies for a packet, the energy expenditure also increases. We must also note that the value of the overhead k has a serious impact on the power consumption. The larger the value of the overhead, the higher the energy expenditure. However, especially for a small value of the overhead k, we can identify an optimal number of simultaneously transmitted packets. Beyond that number, the power consumption metric increases rapidly. This is shown by the slope of the curve: it remains near zero for up to a certain number of simultaneously transmitted packets and then it has a positive value which , consequently, increases the value of the power consumption metric.

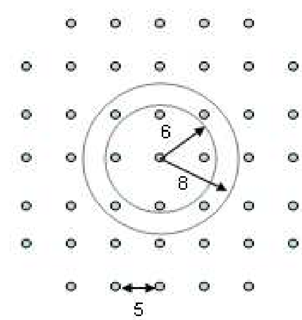

Figure 8. The simulated network topology is a square grid. The distance between neighbouring nodes is 5 units. The number of neighbours a node has depends on its transmission radius.

2.4. Increasing the Communication Range

We now present a different version of the simulation where we vary the transmission radius for each node. The topology of the network remains essentially the same. That is, the nodes are positioned on a plane and they form a square grid. The distance between two adjacent nodes is set to 5 units. Figure 8 depicts the topology of the simulated network. As we can see, the number of neighbours a node has depends on the value of the transmission radius. Increasing this value will allow a node to communicate with more neighbours. We conducted a series of simulations where each time we used another value for the transmission range of the nodes.

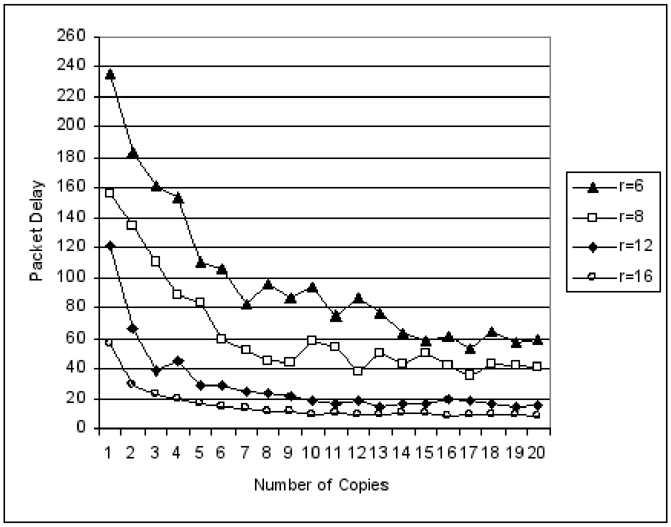

Figure 9. Average packet travel time vs. Number of simultaneously transmitted packets. We illustrate four different cases where the transmission radius r=6, 8, 12 and 16 respectively.

As we can see in Figure 9, increasing the transmission radius of the nodes will decrease the average time a packet needs to reach its destination. When a node has a packet to forward, it will choose one random node among its neighbours. Increasing the numbers of neighbours results in a smaller average packet travel time. Figure 10 illustrates how energy expenditure is affected by different values of the transmission radius. It is clear that when we increase the transmission radius of the nodes, we can achieve lower energy consumption.

3. CONCLUSIONS

In this paper we have simulated the travel time and evaluated the energy consumption of packets in a wireless network. Our results show that the minimum travel time is sensitive to the proper choice of the manner in which packets are destroyed when they have been travelling for an extended period of time. Other factors such as packet loss and variability in paths have a significant effect. Finally we see that the use of multiple copies can be beneficial for both energy consumption and packet travel time.

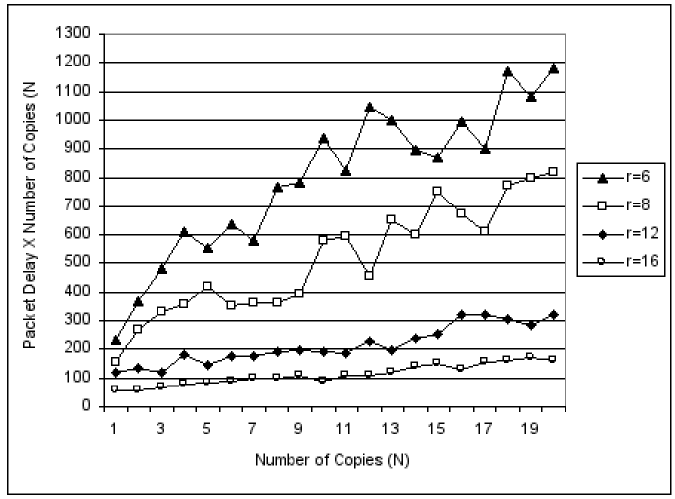

Figure 10. Power Consumption Metric vs. Number of simultaneously transmitted packets. We illustrate four different cases where the transmission radius r=6, 8, 12 and 16 respectively.

REFERENCES

[1] E. Gelenbe, “A diffusion model for packet travel time in a random multi-hop medium,” *To appear in ACM Transactions on Sensor Networks*.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, August 2002.

[3] C. E. Perkins, *Ad Hoc Networking*. New York: Addison Wesley, 2000.

[4] J. Al-Karaki and A. Kamal, “Routing techniques in wireless sensor networks: A survey,” *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6–28, December 2004.

[5] C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed diffusion: A scalable and robust communication paradigm for sensor networks,” in *Mobile Computing and Networking*, 2000, pp. 56–67.

[6] B. Krishnamachari, D. Estrin, and S. Wicker, “Modelling data-centric routing in wireless sensor networks,” in *22nd International Conference on Distributed Computer Systems Workshops (ICDSW02)*, pp. 575–578.

[7] Y.-B. Ko and N. H. Vaidya, “Location-aided routing (LAR) in mobile ad hoc networks,” in *Mobile Computing and Networking*, 1998, pp. 66–75.

[8] E. M. Royer and C.-K. Toh, “A review of current routing protocols for ad hoc mobile wireless networks,” *IEEE Personal Communications*, pp. 46–55, April 1999.

[9] M. Zorzi and R. R. Rao, “Geographic random forwarding (geraf) for ad hoc and sensor networks: Multihop performance,” *IEEE Transactions on Mobile Computing*, vol. 02, no. 4, pp. 337–348, 2003.

[10] S. Shakkottai, R. Srikant, and N. B. Shroff, “Unreliable sensor grids: Coverage, connectivity and diameter.” in *INFOCOM*, 2003.

[11] H. Fotoh, S. Iyengar, and K. Chakrabarty, “Computing reliability and message delay for cooperative wireless distributed sensor networks subject to random failures,” *IEEE Transactions on Reliability*, vol. 54, no. 1, pp. 145–155, March 2005.

[12] E. Gelenbe, M. Gellman, R. Lent, P. Liu, and P. Su, “Autonomous smart routing for network QoS,” in *Proceedings of the First International Conference on Autonomic Computing*, New York, NY, May 2004, pp. 232–239.

[13] E. Gelenbe, R. Lent, and A. Nunez, “Self-aware networks and QoS,” *Proceedings of the IEEE*, vol. 92, no. 9, pp. 1478–1489, Sept. 2004.

[14] E. Gelenbe, R. Lent, and Z. Xu, “Measurement and performance of a cognitive packet network,” *Journal of Computer Networks*, vol. 37, no. 6, pp. 691–701, Dec. 2001.