Solutions to Exam in EL2745 Principles of Wireless Sensor Networks, October 24, 2014

1. Probability of error at the message level

- (a) Rayleigh fading is a statistical model that is often used to describe the effect of a propagation environment on a radio signal, such as that used by wireless sensors. It is most applicable when there is no dominant propagation along a line of sight between the transmitting and receiving node. Because there is no direct ray component, Rayleigh fading is often classified as the worst case fading type. Rayleigh fading models assume that the complex envelope of the received signal is the sum of many random complex components arriving from different paths and its amplitude follows the Rayleigh distribution.
- (b) Let $P(\gamma)$ be the probability of error for a digital modulation as a function of E_b/N_0 , γ , in the Gaussian channel. Let the channel amplitude be denoted by the random variable α , and let the average SNR normalized per bit be denoted by $\gamma^* = \mathbf{E}[\alpha^2]\mathbf{E_b}/\mathbf{N_0}$. Then to obtain P(e) for a Rayleigh fading channel, $P(\gamma)$ must be integrated over the probability that a given γ is encountered:

$$P(e) = \int_0^\infty P(\gamma) p(\gamma) d\gamma,$$

For Rayleigh fading,

$$p(\mathbf{\gamma}) = \frac{1}{\mathbf{\gamma}^{\star}} e^{-\mathbf{\gamma}/\mathbf{\gamma}^{\star}}$$

In the case of coherent BPSK, the integration can actually be computed yielding

$$P(e) = \frac{1}{2} \left[1 - \sqrt{\frac{\gamma^{\star}}{1 + \gamma^{\star}}} \right]$$

At high SNR such as OQPSK systems, the approximation $(1+x)^{1/2} \sim 1+x/2$ can be used, giving

$$P(e) \sim rac{1}{4\gamma^{\star}}$$

compared with $P(e) = \mathbf{Q}(\sqrt{2\gamma^{\star}})$ for the Gaussian channels.

(c) Given the bit error probability P(e), the probability of successfully receiving a message is

$$p = (1 - P(e))^f.$$

For Rayleigh fading, from previous section we have

$$P(e) \approx rac{1}{4\mathrm{SNR}}$$
.

(d) To have a message reception probability of at least p > 0.35, it is required that

$$\left(1 - \frac{1}{4\mathrm{SNR}}\right)^f > 0.9^{10}$$

By substituting f = 10, and

$$\mathrm{SNR} = \frac{\alpha E_b}{N_0 d^2} = \frac{10}{d^2}$$

in inequality above we obtain

$$\left(1-\frac{d^2}{40}\right)^{10} > (0.9)^{10} \Longrightarrow d \le 2.$$

2. Analysis of CSMA based MAC in WSNs

- (a) The protocol is supported in the "beacon-less" modality of IEEE 802.15.4.
- (b) The sensors nodes use slotted CSMA scheme with fixed contention size M. Nodes sense the channel and if it is free they enter to the contention round, where each node draws a random slot number in [1,M] using uniform distribution and sets its counter with this integer number. In successive slots times of duration t_{slot} each contender counts down until when its counter expires then it senses the channel and if there is no transmission in the channel it will send the packet immediately at beginning of the next slot. Assume t_d is the required time to transmit the data packet. t_{slot} is determined by physical layer parameters like propagation time of the packet (it also called vulnerable time) which is defined by the distance between the nodes. Each contention round will finish by a packet transmission that might be either successful or collided. Collision happens if at least two nodes draw the same minimum slot number, otherwise the transmission would be successful.
- (c) Consider the contention round with length M and total number of sensors N. Let x_n be the selected slot of node n. Let $p_s(m)$ be the probability of having a successful transmission at slot m which happens when a node selects slot m and rest of the nodes select greater slots. P_s is the probability of success over the entire contention round and is obtained by the summation over $p_s(m)$

$$P_{s} = \sum_{m=1}^{M} p_{s}(m) = \sum_{m=1}^{M} \sum_{n=1}^{N} \operatorname{Prob}\{x_{n} = m, x_{j} > m \; \forall j \neq n\}$$
$$= \sum_{m=1}^{M} {N \choose 1} \frac{1}{M} \left(1 - \frac{m}{M}\right)^{N-1}$$

(d) Let $p_c(m)$ be the probability of collision at slot *m* then

$$p_c(m) = \operatorname{Prob}\{x_n \ge m, \forall n\} \cdot \left[1 - \operatorname{Prob}\{x_n = m, x_j > m \forall j \neq n | x_n \ge m \forall n\}\right]$$
$$-\operatorname{Prob}\{x_n \ge m+1 | x_n \ge m, \forall n\} \left]$$
$$= \frac{1}{M^N} \left[(M-m+1)^N - (N+M-m) \cdot (M-m)^{N-1} \right],$$

which is essentially one minus the probability of having successful or idle slots. Also the probability of having collision after contention round can be formulated in a similar way as success case. i.e.,

$$P_c = \sum_{m=1}^{M} p_c(m) = 1 - P_s.$$

3. Shortest path routing in WSNs



Figure 1: A sample topology of the WSN. Node 1 is the sink and link qualities(PRR) are depicted on each arcs

(a) Given the PRR of each link, the mean transmission number required per successful packet delivery on the link (i, j) is

$$E[TX] = \sum_{t=1}^{\infty} t \text{PRR}(i, j)^{t} (1 - \text{PRR}(i, j))^{t-1} = \frac{1}{\text{PRR}(i, j)}$$

(b) Denote $\text{ETX}[x_i]$ as the expected number of transmissions required for node x_i to send a packet to the sink. Also, denote \mathcal{N}_i and \mathcal{P}_i as the neighbors set and parent of node *i*, respectively. Then given PRR(i, j)as the packet reception rate from *i* to *j*, One can formulate $\text{ETX}[x_i]$ as

$$\operatorname{ETX}[x_i] = \min_{j \in \mathcal{N}_i} \left\{ \operatorname{ETX}[x_j] + \frac{1}{\operatorname{PRR}(i,j)} \right\}$$

and $\mathcal{P}_i = \{x_j\}$ where x_j is the neighbor that minimizes the ETX $[x_i]$. ETX $[x_1] = 0$ where x_1 is the sink. Starting from the sink, nodes put their ETX equal to infinity. Then sink propagates its ETX value to one hop neighbors, they update their ETX and broadcast their values. Whenever a node receives a ETX message from a neighbor, it checks if the value differs from the previous reported value. If so they update their ETX and parent node (if it happens) and broadcast their ETX. It can be proved that this algorithm constructs a MST and converges in a couple of iterations (as long as PRR values remain unchanged).

- (c) Denote by *P* the set of nodes whose shortest path to the destination node is known, and denote by D_j the current shortest distance from node *j* to the destination node. Note that only when node *j* belongs to the set *P* can we say D_j is the true shortest distance. Choose node 1 as the destination node. Initially, set $P = \{1\}$, $D_1 = 0$, and $D_j = \infty$ for $j \neq 1$.
 - (i) Update D_j for $j \neq 1$ using the following equation

$$D_j = \min[D_j, d_{j1}].$$

(ii) Find *i* such that

$$D_i = \min_{j \notin P} [D_j]$$

update $P := P \cup \{i\}$.

(iii) Update D_j for $j \notin P$ by the following equation

$$D_j := \min[D_j, D_i + d_{ji}]$$

in which *i* is the *i* obtained in (ii).

- (iv) Go back and compute steps (ii) and (iii) recursively until P contains all the nodes in the network. The resulting D_j is the shortest distance from node j to node 1.
- (d) According to Figure 1 nodes update their ETX as following

$$ETX[1] = 0$$

node 3:

$$\operatorname{ETX}[3] = \min\left\{\frac{1}{0.9}, 1 + \operatorname{ETX}[2]\right\} = \min\{1.1, \infty\} = 1.1, \mathcal{P}_3 = \{1\}.$$

node 2:

$$\operatorname{ETX}[2] = \min\left\{\frac{1}{0.8}, 1 + \operatorname{ETX}[3]\right\} = \min\{1.25, 2.1\} = 1.25, \mathcal{P}_2 = \{1\}.$$

Note that here we assumed node 2 receives the ETX[3] before computing its value.

node 4:

$$ETX[4] = \min\left\{\frac{1}{0.5} + ETX[3], \frac{1}{0.6} + ETX[5], \frac{1}{0.7} + ETX[6]\right\}$$
$$= \min\{3.1, \infty, \infty\} = 3.1, \mathcal{P}_4 = \{3\}.$$

node 5:

$$ETX[5] = \min\left\{\frac{1}{0.8} + ETX[2], \frac{1}{0.6} + ETX[4], \frac{1}{0.5} + ETX[6]\right\}$$
$$= \min\{2.5, 4.77, \infty\} = 2.5, \mathcal{P}_5 = \{2\}.$$

node 6:

$$ETX[6] = \min\left\{\frac{1}{0.7} + ETX[4], \frac{1}{0.5} + ETX[5]\right\}$$
$$= \min\{4.53, 4.5\} = 3.5, \mathcal{P}_6 = \{5\}.$$

in next iteration all the ETX values will remain unchanged and the algorithm converges. The set of \mathcal{P} 's builds the topology.

4. Collaborative multilateration

(a) Solve the equations

$$x^{2} + (y-3)^{2} = (3.5 - \Delta)^{2},$$

$$(x-4)^{2} + y^{2} = (4.5 - \Delta)^{2},$$

$$(x-4)^{2} + (y-3)^{2} = (5.5 - \Delta)^{2}.$$

We have the position of U is (0,0) with $\Delta = 0.5$.

- (b) From the problem, we see that both U and V are placed on a common axis. Then we have U = (0, 0.5), V = (0, -0.5) directly.
- (c) Clearly $\hat{U}_0 = (0, 1)$ and $\hat{V}_0 = (0, -1)$.
- (d) The squared distances from *U* to nodes *A*, *C*, and *V* are respectively 1.25, 1.25, and 1, while the squared distances from *V* to *B*, *D*, and *U* are 1.25, 1.25, and 1. Then for the first calculation we have $r_A = r_C = 1$ and $r_v = 2$ so that

$$A = \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & -2 \end{bmatrix}, \qquad z = \begin{bmatrix} 1 - \sqrt{1.25} \\ 1 - \sqrt{1.25} \\ 2 - \sqrt{1} \end{bmatrix}.$$

resulting in the system $A^T A d_u = A^T z$:

$$\begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} \delta_{xu} \\ \delta_{yu} \end{bmatrix} = \begin{bmatrix} 0 \\ -2 \end{bmatrix},$$

which gives $\hat{U}_1 = (0, 1 - 0.5) = (0, 0.5)$. In the next set, $r_B = r_D = 1$, $r_U = 1.5$, so that

$$A = \begin{bmatrix} 1 & 0\\ -1 & 0\\ 0 & 1.5 \end{bmatrix}, z = \begin{bmatrix} 1 - \sqrt{1.25}\\ 1 - \sqrt{1.25}\\ 1.5 - \sqrt{1} \end{bmatrix}.$$

The result is $\hat{V}_1 = (0, -1 + 0.333) = (0, -0.667)$. Iterations can continue now for \hat{U}_2 using *A*, *B*, and \hat{V}_1 and so forth. Successive iterations produce results closer to the true ones. In general, for collaborative multilateration to converge a variety of constraints on the topology must be satisfied, and the order of the computations is important. However, if there is a relatively high density of nodes with known position these constraints are almost always satisfied without explicit checking being required; bad positions can be discarded through recognition that the values are diverging in some neighborhood.

5. Networked Control System

(a) Since $\tau < h$, at most two controllers samples need be applied during the k-th sampling period: u((k-1)h) and u(kh). The dynamical system can be rewritten as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), & t \in [kh + \tau, (k+1)h + \tau) \\ y(t) &= Cx(t), \\ u(t^+) &= -Kx(t - \tau), & t \in \{kh + \tau, k = 0, 1, 2, ...\} \end{aligned}$$

where $u(t^+)$ is a piecewise continuous and changes values only at $kh + \tau$. By sampling the system with period *h*, we obtain

$$x((k+1)h) = \Phi x(kh) + \Gamma_0(\tau)u(kh) + \Gamma_1(\tau)u((k-1)h)$$

$$y(hk) = Cx(kh),$$

where

$$\begin{split} \Phi &= e^{Ah} = e^{ah} \,, \\ \Gamma_0(\tau) &= \int_0^{h-\tau} e^{As} B ds = \frac{b}{a} \left(e^{a(h-\tau)} - 1 \right) \,, \\ \Gamma_1(\tau) &= \int_{h-\tau}^h e^{As} B ds = \frac{b}{a} \left(e^{ah} - e^{a(h-\tau)} \right) \,. \end{split}$$

given that A = a, B = 1, C = 1.

(b) Let $z(kh) = [x^T(kh), u^T((k-1)h)]^T$ be the augmented state vector, then the augmented closed loop system is

$$z((k+1)h) = \tilde{\Phi}z(kh),$$

where

$$ilde{\Phi} = \left[egin{array}{cc} \Phi - \Gamma_0(au) K & \Gamma_1(au) \ -K & 0 \end{array}
ight]$$

Using the results obtained in (a), we can obtain

$$ilde{\Phi} = \left[egin{array}{c} e^{ah} - rac{b}{a} \left(e^{a(h- au)} - 1
ight) K & rac{b}{a} \left(e^{ah} - e^{a(h- au)}
ight) \ -K & 0 \end{array}
ight]$$

(c) The characteristic polynomial of this matrix is

$$\lambda^2 - \left(e^{ah} - \frac{b}{a}\left(e^{a(h-\tau)} - 1\right)\right)K + \frac{Kb}{a}\left(e^{ah} - e^{a(h-\tau)}\right).$$

Thus when the max $|\lambda| > 1$, the closed loop system becomes unstable.

(d) We use the following result to study the stability of the system:

Theorem 1 Consider the system given in Fig. 2. Suppose that the closed-loop system without packet losses is stable. Then

- *if the open-loop system is marginally stable, then the system is exponentially stable for all* $0 < r \le 1$.
- *if the open-loop system is unstable, then the system is exponentially stable for all*

$$\frac{1}{1-\gamma_1/\gamma_2} < r \le 1\,,$$

where
$$\gamma_1 = \log[\lambda_{max}^2(\Phi - \Gamma K)]$$
, $\gamma_2 = \log[\lambda_{max}^2(\Phi)]$

Here we have

$$\Phi = e^{Ah} = e^{ah},$$

$$\Gamma = \int_0^h e^{As} B ds = \frac{b}{a} \left(e^{ah} - 1 \right)$$

Thus, the stability of this system depends on the values of K, h, a. When the conditions are not satisfied, from a control theory point of view we may choose different K for controller, or different sampling time h for the system to make the system stable. Instead, from a networking point of view, we may change the protocol parameters to so have a packet loss probability that meets the stability conditions.