1 Frame synchronization and Symbol timing recovery at baseband

Sending a sequence of bits as frames

1. Generate a random sequence of 1000 data bits.

2. We need to model sending this random sequence of 1000 bits as frames. Each frame consists of 50 data bits and a 13 bit header which is chosen to be a 13 bit Barker sequence. Find out what is a Barker sequence and write a matlab function which can convert a sequence of data bits into frames in the above format.

3. We now need to simulate transmission of the baseband waveform corresponding to the frames that you have generated above.

4. Assume that each bit in a frame is transmitted for a time T_b. Also assume that consecutive frames are transmitted with a time gap of $D \times T_b$ or D bit times, where D is a discrete uniform random variable taking values in $\{10, 11, 12, \cdots, 100\}$. Assume that there is a random gap of D before the first frame is sent. Note that D is sampled every time before a frame is sent. A bit 1 is transmitted as a pulse of amplitude A and a bit 0 is transmitted as a pulse of amplitude $-A$.

5. Obtain the baseband waveform corresponding to the sequence of frames that you have obtained above. Note that the baseband waveform has to be a sampled representation of the actual continuous time signal.

Frame synchronization using autocorrelation

1. The objective in this subtask is to estimate/obtain the start of each frame from the received waveform.

2. Simulate the reception of the above baseband waveform transmitted over an ideal channel which has an attenuation γ (constant for all frequencies) and introduces sampled Gaussian noise of variance σ^2. Please note that the channel should not distort the waveform (use a low pass filter of sufficiently high bandwidth or even an all-pass filter). Please use a matched filter to receive the above baseband waveform.

3. Note that because of the way the transmitted waveform is constructed and the channel (model) over which the waveform is transmitted, sampling the received waveform uniformly at multiples of T_b gives you samples from different bit times. However note that during the spaces between the samples (during a sample of D, you might be sampling from noise). So obtain a sequence of samples Z_n from the received waveform by sampling at multiples of T_b.

4. Compute the autocorrelation of the sequence Z_n with the Barker sequence, what do you observe?

5. Write down a function (in Matlab) which computes the autocorrelation of the Barker sequence with a 13 length moving window of the received sequence. The function should return an autocorrelation value every iteration.

6. Can this function be used to estimate the start of the frames? Devise a decision rule that can be used to detect the start of each frame.

7. Obtain the fraction of frames for which the start time is estimated in error. Call this the frame-start-error-rate.

8. Repeat the above sequence of tasks for a case where 1000 frames are transmitted, i.e., you should start with 1000×50 bits.

9. Now plot the frame-start-error-rate as a function of σ^2 and write down your observations.

Early late gating - 1

1. We will first simulate a single pulse which is transmitted over a baseband channel arriving at the receiver after a random unknown propagation delay. The received pulse can simply be modelled as a pulse waveform with a random initial shift.

2. Suppose we use a matched filter for receiving and further processing of these pulses. Note that since the random initial shift is not known to the receiver, the receiver samples the output of the matched filter at some time, which it assumes is that time at which the output is maximum.

3. Using Early-Late gating obtain the sampling instant for the case of the single pulse. Note that multiple iterations can be carried out on the single pulse itself to find out this sampling instant.

Early late gating - 2

1. We will consider a slightly different early-late gating method which is applied on a sequence of bits, rather than a single bit as discussed above.

2. Generate a sequence of alternating 0s and 1s of length N.

3. Convert the above sequence of bits into a baseband waveform (sampled)

4. Pass the above baseband waveform through a channel with introduces sampled Gaussian noise with variance σ^2, attenuation γ, and a random delay D (choose D uniformly in the number of samples in $[0, T_b]$).

5. Note that when the receiver starts, it does not know the random delay D. Therefore it might start sampling the received signal at T_b - not accounting for the random delay D.

6. Visualize the received signal after matched filtering using an eye diagram and observe the sample values at the receiver's estimate of the sampling time.

7. Now run the early-late gating method that you have developed in the previous task here, but run it not on the first bit period T_b, but on multiple bit periods.

8. After N bit times, compute the squared error between the receivers estimate of the sample time and the ideal sample time. Note that this squared error has been obtained for a fixed σ^2 and γ.

9. Repeat the above sequence of steps for 1000 times (generate different samples of D) and obtain the mean of the 1000 squared error quantities.

10. Repeat for different values of σ^2 and plot the mean squared error as a function of σ^2.

11. Repeat for different values of N, keeping σ^2 constant and plot the mean squared error as a function of N.

Symbol and frame synchronization

1. In this task, we will attempt to do symbol and frame synchronization together. We note that in the "frame synchronization using autocorrelation" task, symbol synchronization was assumed since the gaps between the frames are multiples of T_b. Here we will consider the case where the channel introduces a random delay and the gaps between frames are not restricted to be multiples of T_b.

2. Again generate a random sequence of 1000 data bits.

3. We need to model sending this random sequence of 1000 bits as frames. Each frame consists of 50 data bits and a 23 bit header which is chosen to be a 10 bit sequence of alternating 1s and 0s and then a 13 bit Barker sequence. Write a matlab function which can convert a sequence of data bits into frames in the above format.

4. We then simulate transmission of the baseband waveform corresponding to the frames that you have generated above.

5. Assume that each bit in a frame is transmitted for a time T_b. Also assume that consecutive frames are transmitted with a time gap of D samples, where D is a discrete uniform random variable taking values in $\{0, 1, 2, \cdots, 50\}$.

6. Also assume that there is a random gap of D before the first frame is sent. Note that D is sampled every time before a frame is sent. A bit 1 is transmitted as a pulse of amplitude A and a bit 0 is transmitted as a pulse of amplitude $-A$.

7. Obtain the baseband waveform corresponding to the sequence of frames that you have obtained above. Note that the baseband waveform has to be a sampled representation of the actual continuous time signal.

8. Pass the above baseband waveform through a channel with introduces sampled Gaussian noise with variance σ^2, attenuation γ, and a random delay D (as above).

9. The received signal is matched filtered, and applied to the early late gating (2) method that you have developed.

10. The sampling time estimate is used to obtain the samples Z_n again.

11. Now obtain the frame start times using the method that you have used in frame synchronization task.

12. The performance of the synchronization method is evaluated by finding out the average number of frames which are received in error.Call this the frame-start-error-rate.

13. Repeat the above sequence of tasks for a case where 1000 frames are transmitted, i.e., you should start with 1000×50 bits.

14. Now plot the frame-start-error-rate as a function of σ^2.