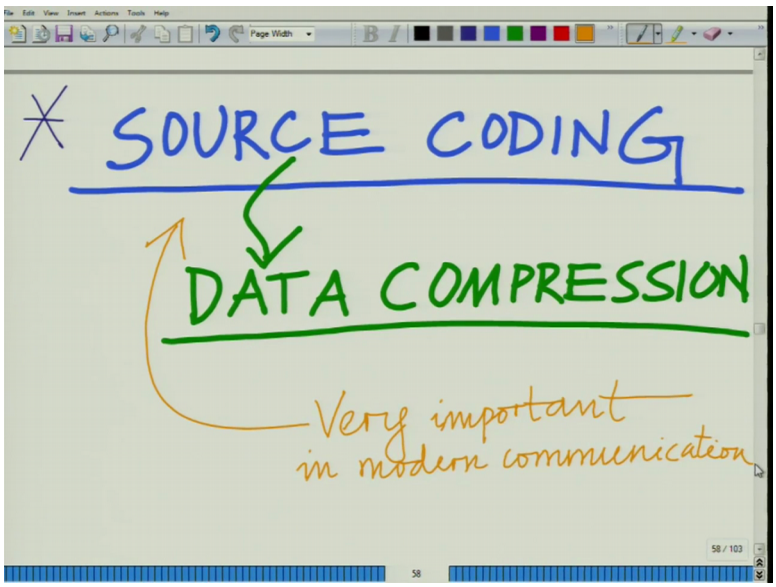
Principles of Communication Systems - Part II
Prof. Aditya K. Jagannatham
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

Lecture - 41
Introduction to Source Coding and Data Compression, Variable Length Codes, Unique Decodability

Hello. Welcome to another module in this massive open online course. So far we have looked at various interesting results in information theory, various applications of information theory more in fundamentally, we have looked at the channel capacity of various channels and information theory. Let us change tracks now and look at another aspect or another important application of information theory and that is towards source coding, ok.

(Refer Slide Time: 00:39)

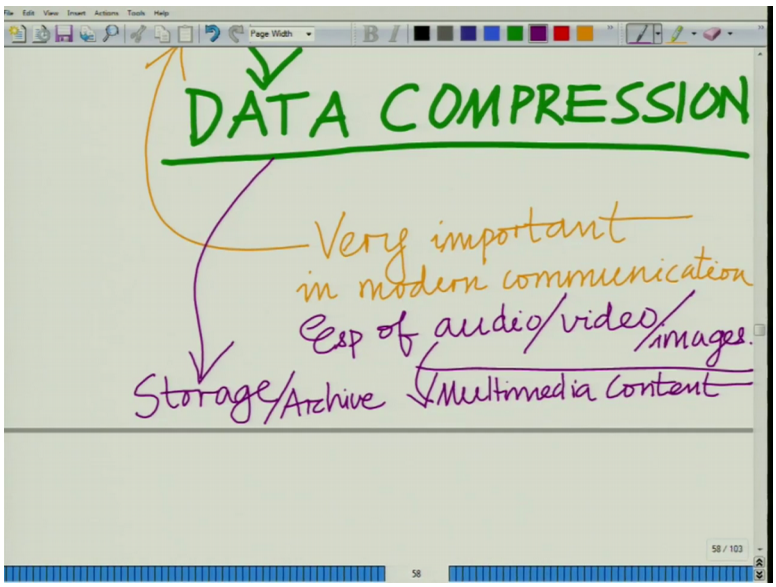


So, we are going to briefly look at techniques for source coding and what we do in source coding is basically, this can also be thought of as source coding or basically, the aim is towards compressing the data. So, you can also be thought of a source coding. So, if you look at most of the modern compression standards for instance, what makes communication possible at the high data rate and modern wireless communication systems is the development of modern communication technologies, but together with that what make communication of for instance, not just audio and data.

But what makes communication of a rich multimedia contents such as audio and video, right audio images and video possible is also the development of efficient compression takes because if you look at a typical audio file or a video file contents, a large amount of the raw file contents, a large amount of data which has to be appropriately compressed, so that it can be transmitted over the communication channel which has a practical bandwidth constraint correct. So, these communicates data compression of source coding technologies or source coding techniques which in turn are motivated and justified. Information theory have played as much a role in the development of modern communication techniques as have been played by the development of modern communication wireless communication technologies, ok.

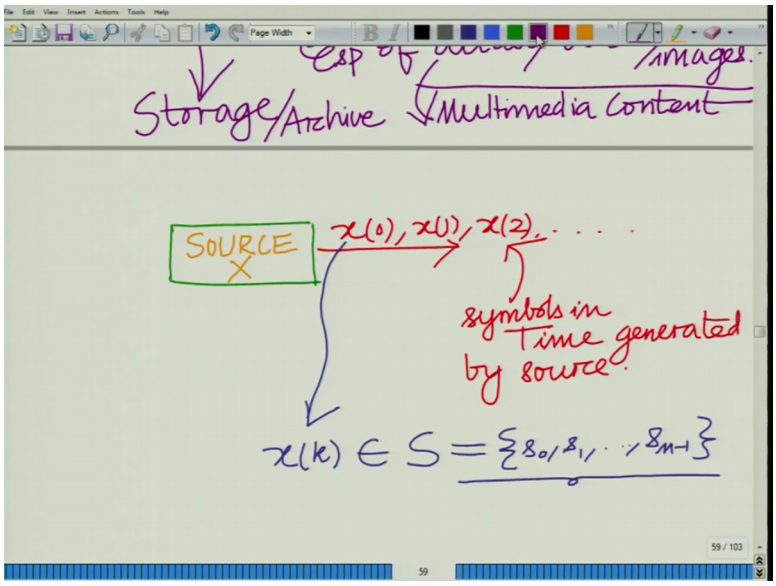
So, this is very important in modern communication, especially multimedia content, such as audio, video, images etcetera, correct.

(Refer Slide Time: 03:00)



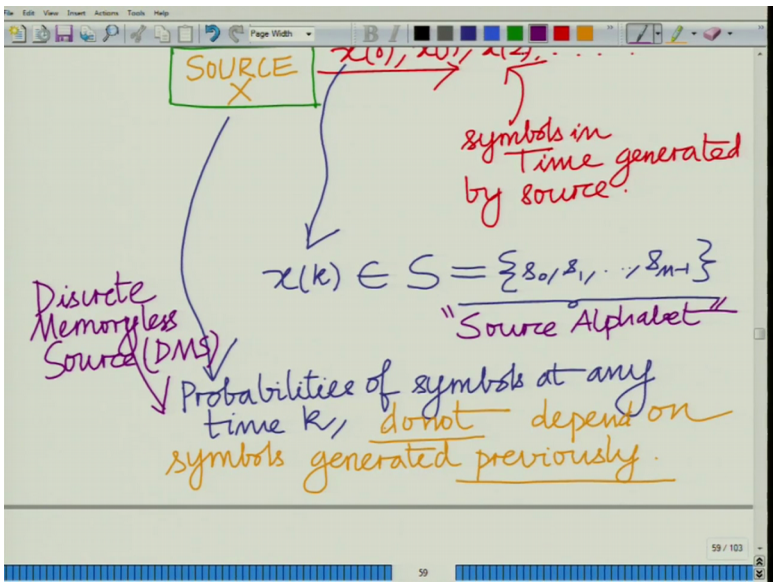
Basically we can think of this together and not just in communication, but also storage to minimize the storage content to generate to store the huge amount of multimedia to store and archive to store and archive the huge volumes of multimedia data say in the form of video and images, currently being generator or the internet and several applications is important to efficiently store them. So, compression place a very important role therefore towards storage and archival of this large amounts of multimedia content that is being generated, all right.

(Refer Slide Time: 04:14)



Now, to put simply we have changing tracks from channel coding to looking at source coding and to put simply, what source coding implies is the following that is if you look at source, let say you call it x and this generates various symbols for instance x 0 x 1. So, these are the various symbols in time that are generated by the source and now, the source symbols of course as we have seen before, each source symbol for instance x k is the symbol at time instant k can be thought of as belonging to the set fix set of symbols s0 s1 up to sm minus 1.

(Refer Slide Time: 05:33)

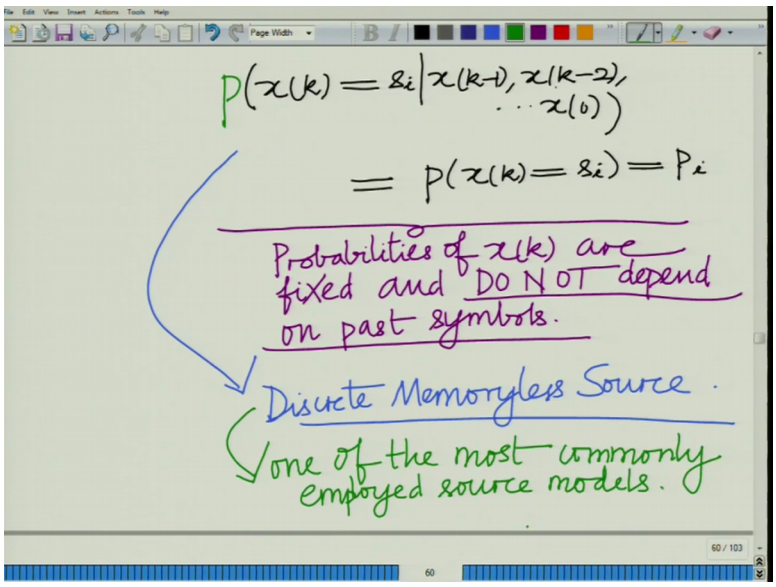


So, symbols are generated from, we also said is basically the source alphabets or the symbols are generated from source alphabet comprising of symbols alphabet comprises of the symbols s naught s 1 up to m minus 1.

So, as a source alphabet of size m, this is something that we have already seen before that is, we have a source which is generating these symbols. Now, let us define another important kind of source also. We have not paid. We are not accurately defined it. Previously this is an assumption which was implicit in all the analysis that was being carried out thus far is that these different symbols are independent identically distributed, that is what we mean is that since we are sources generating several symbols in time, what we mean is that the symbol that is probabilities of symbol of various symbols at it anytime k do not depend on the symbols that are generated previously, such a source is termed as a discrete. This is also known as a discrete memory less source.

So, what do we mean by this discrete memory less source? That is what we are saying is that if you look at any symbol x k generated at time instant k, the probability that x x k is s naught does not depend on what are x k minus 1, x k minus 2 and so on. It does not depend on any of the previous symbols, correct. So, each of these symbols are generated in an independent identically distributed fashion, such as source is known as a discrete memory less source to qualified or to put mathematically.

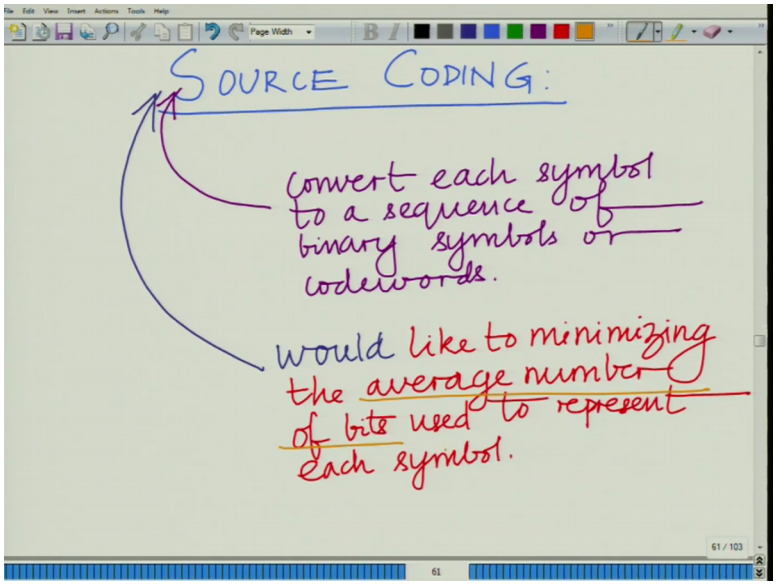
(Refer Slide Time: 08:33)



What we mean is basically although it is very intuitive, this notion is very intuitive. That is the reason that p of x k equals s i conditioned on x k minus 1 x k minus 2 and so on and were given take a number of past symbols that is so on up to x0. This is simply equal to the probability x k equal to s i equals a fixed quantity equals of fixed quantity p i, ok.

So, the probabilities independent, the probabilities are fixed. So, what you can see from here is that probability of each symbol are fixed and do not depend on the past symbols and such a source is known as a discrete memory less source and this is one of the most commonly employed source models. This discrete memory less source is one of the most commonly employed source models. Now, let us come to the source model. We are going to consider a discrete memory less source which is one of the simplest models that can be employed to model sources which is also one of the most commonly occurring and one of the most frequently employed models to modulus source, all right. It is frequently applicable in practicing since several sources, one would encounter can be closely modeled, can be either directly modeled or closely approximated as discrete memory less sources.

(Refer Slide Time: 11:37)

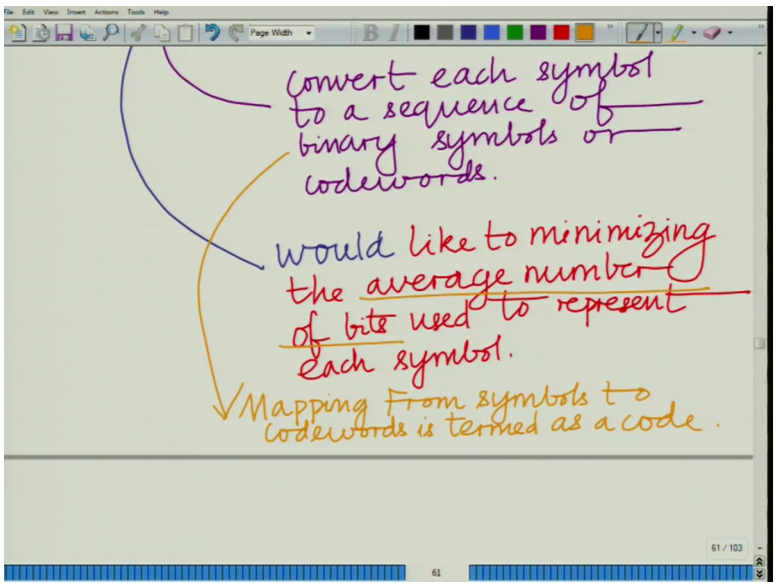


Now, let us come to the central theme that is source coding. Now, what do we do in source coding? We convert each symbol to a sequence of binary symbol. So, let me write this as done. We convert each symbol to a sequence of binary information; binary symbols or binary bits let say or code. So, this sequence of binary symbols binary bits is

also termed as a codeword. So, each symbol is basically converted into codeword represented in terms of binary information bits, ok.

Now, what is our aim in source coding? Obviously, remember we said its use towards data compression, so we want to minimize the number of bits, all right intuitively. Obviously, the smaller the number of bits, the more easier it is to communicate over a channel, the more easier it is to source. So, we would like to minimize the number of bits and in particular a useful metric is minimize the average number of bits that is used to represent each symbol. So, rather than looking at each symbol and isolation, we would like to minimize the average number of bits that can be used to represent the symbol. So, we would like to minimize or we would like to strive towards minimizing the average number of bits to represent and this is the average number of bits used to represent. So, we are to represent it in a compact fashion, correct and this mapping from symbols to code.

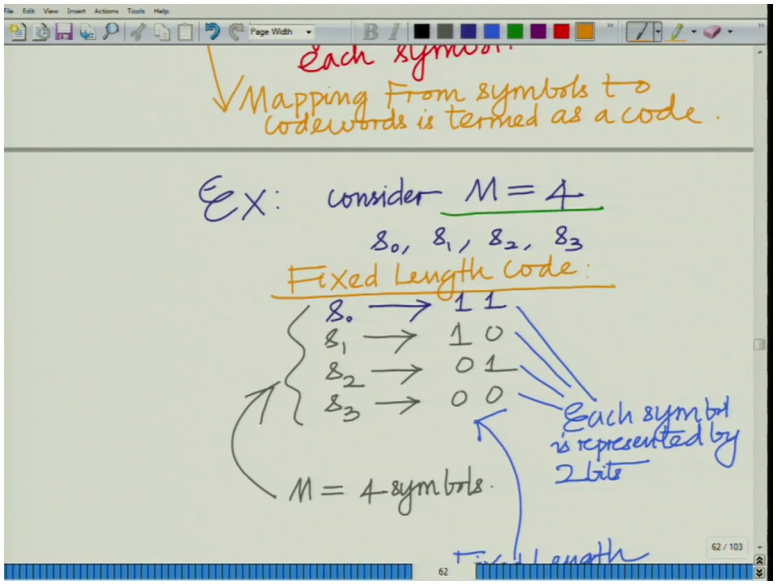
(Refer Slide Time: 14:51)



So, this mapping from the symbols to code is symbols to codeword is codewords. So, this mapping from symbols to codeword this is what is termed as a code, all right. So, we would like to construct an efficient code which maps the symbols to the codewords in such a fashion that average number of bits, the average length of each codeword used to represent a symbol is minimized and naturally, there are different kinds of codes that are

possible, different categories of code constructions. Let us look at some of the most characteristics, some of the most popular ones, ok.

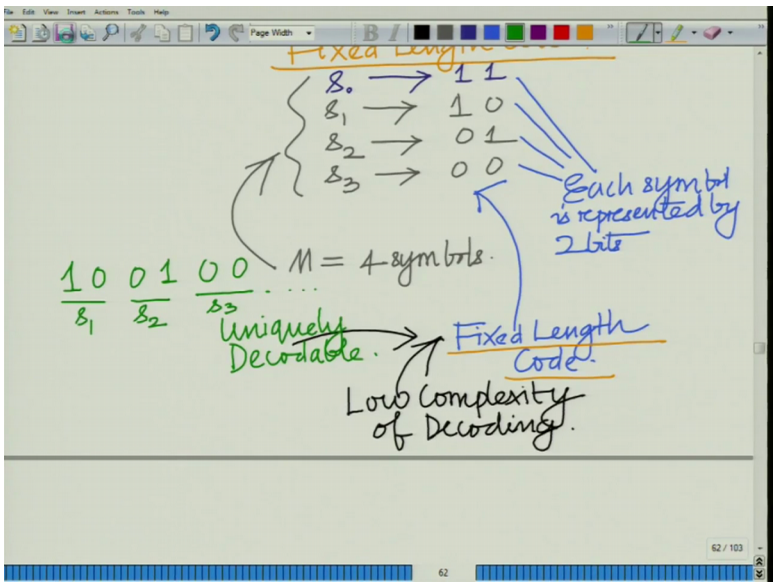
(Refer Slide Time: 15:47)



So, for example, let us look a typical code. Let us say M is equal to 4 that is the size of the source alphabet is 4, source alphabet is of size 4. Therefore, we have s0, s1, s2, s3.

Now, let us say s0 is represented using 1, 1. S1 is represented using 1, 0, s2 is represented using 0, 1 and s3 is represented using 0, 0.

(Refer Slide Time: 16:44)

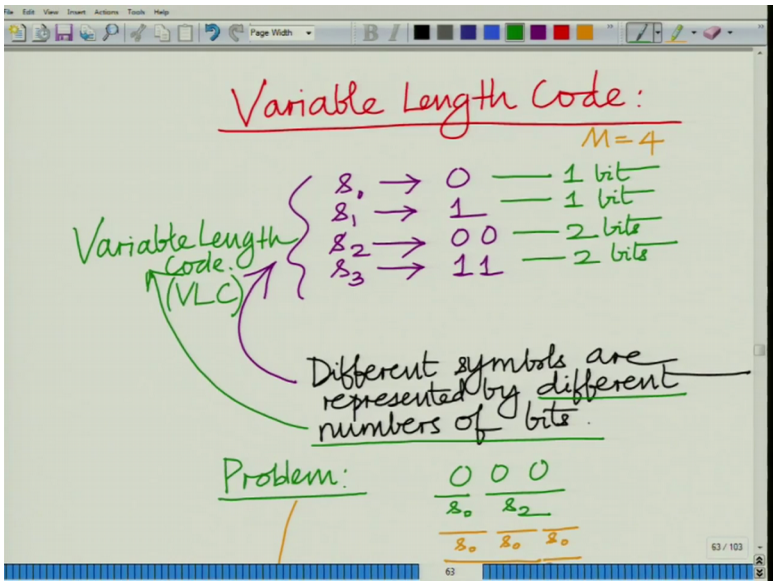


So, we have four symbols. We are representing each code or each symbol is represented using two bits and therefore, the number of bits is 2 for each symbol and this is termed as a. So, this mapping is what is the code and this is also of fixed length code. You can see that this is a very specific code in which each symbol is represented using a fixed number of, all the symbols are used to represent in the same number of bits which in this case is 2. So, this is fixed. All it is not necessary that all symbols be represented using the same number of bits. In fact, we will see that for optimality, this is not generally true that is to get the code which represents each symbol using the lowest number of bits which represents, which minimizes the average number of symbols for bit, all right.

So; however, this code which represents each symbol using in which all symbols are represented using the same number of bits is termed as a fixed length code and decoding of fixed lengths code is very simple because you know each symbol corresponds to the fixed set of bits. Let us say the number of bits is 2. So, you take the first 2 bits, decode the symbol; next 2 bits, decode the symbol; next 2 bits and decode the symbol. So, this kind of a code is uniquely decodable. These are the terms that we are going to keep introducing uniquely decodable in the sense that given the bits, I can uniquely reconstruct the transmitted symbols. So, let us note here that fixed length code is very simple or like might not be efficient. It has a low complexity of decoding.

Further, it will also be uniquely decodable. We will look at this term a little bit later what we mean by uniquely decodable, but you can see roughly it means that given a set of code bits, I can uniquely reconstruct the symbols that have been transmitted. For instance, if we have 1 0 0 1 0 0, now you look at the first 2 bits, we know its a fixed length code. So, 1 0 will be s1, 0 1 will be s2, 0 3 and the s3 and so on. So, this is unique. So, it is very simple and it is uniquely decodable. So, you look at groups of 2 bits each starting from the first bit. So, you take the 2 bits decoded, next two bits decode and keep progressing and so on. Now, let us next come to a naturally we have fixed length code, obviously if it is not a fixed length code, then it is known as a variable length code.

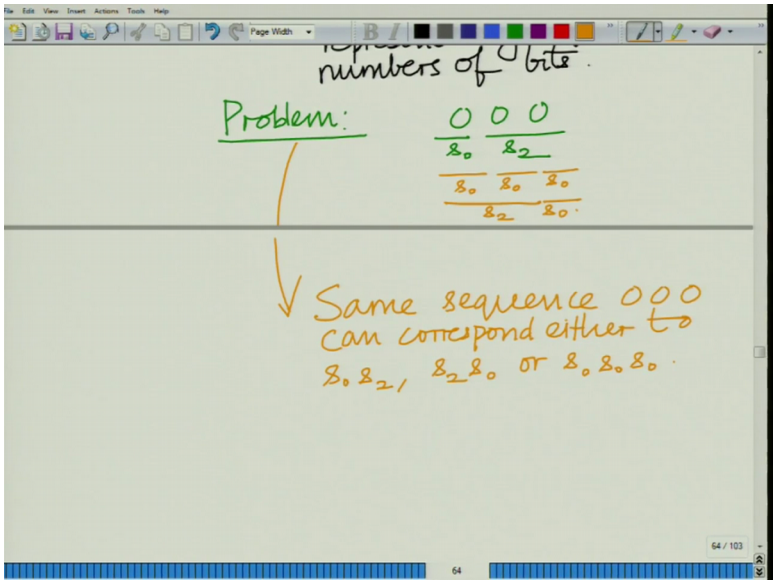
(Refer Slide Time: 20:24)



Now, in a variable length code for instance, again let us take an example of M equal to 4. For instance, we have s0. Let us take an example s0 represented by 0, s1 represented by 1, s2 represented by 0 0, s 3 represented by 1 1, ok.

Now, the variable length code, obviously you can see different symbols are represented using different numbers of bits and further observed in this case, there is a problem. It is a variable length code. For instance, here we have for 0, we have 1 bit. This is 1 bit, 2 bits and 3 bits. Now, there is a problem for instance, what is the problem?

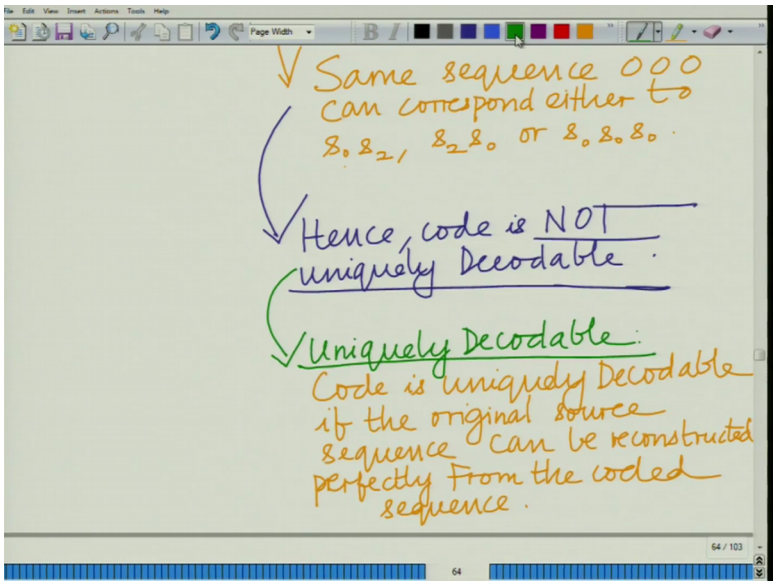
(Refer Slide Time: 22:28)



Let us look at the problem and the problem should be apparent to you. Here if you look at the transmitted sequence 0, either correspond to s0 followed by s2 or it can correspond to s0 s0 s0 or it can correspond to s2 followed by s0, ok.

So, the same sequence 0 0 0, you can correspond to either s0 s2 s0 s0 s0 or s2 s0. Note this problem. So, same sequence what is the problem here in this example, the same sequence can correspond either to s0 s2 s2 s0 or s0 s0. So, these three sequences, given the received bits sequence 0 0 0, either the stored bit sequence or the bit sequence 0 0 0 at the receiver, one cannot uniquely decode the transmitted symbol sequence or the transmitted source coded sequence. It can either correspond to s2 s0 s0 s2 or s0 s0 s0.

(Refer Slide Time: 24:26)



So, this code is not uniquely decodable. So, this code, hence code is not uniquely decodable. So, original sequence what do we mean by the original sequence? What do we mean by uniquely decodable? Let us now formally define what we mean by, we see a code is uniquely decodable. If the original symbol sequence is uniquely decodable if the original if the original source sequence or the original source symbol sequence, original source sequence can be reconstructed perfectly from the coded sequence. In this case, we have seen that there exist at least 1 that is 0 0 0 at corresponding to which the symbol sequence, the corresponding symbol sequence cannot be decoded uniquely. There are multiple symbol sequence which give raise to the same code sequence, right so many to one mapping.

So, not one to one mapping and therefore, this code is not uniquely decodable which creates problems for us because either when the symbols have been received at the receiver or if the symbols have been stored and you try or they have the bits have been stored and you are trying to reconstruct the original symbol sequence, then it cannot be the symbol sequence is unknown. It cannot be reconstructed uniquely, all right. So, there is an information loss that happens in the source coding process and naturally this is not preferable. So, therefore, what we are going to do? So, what we have seen here is, we have seen now that does not mean that all variable length codes are not uniquely decodable here.

We have considered a variable for instance, therefore symbols are different numbers of bits. So, this is a variable length code or what is also termed as VLC. Sometimes now of course, it is clear that for a fixed length code is always going to be uniquely decodable, correct as long as many symbols are not mapped to the same fix length codeword, the fix length code because its nature is uniquely decodable, correct. How are the variable length code? This is not the case. Now, what are the conditions to be satisfied by a code, variable length code? For that matter, any code for it to be uniquely decodable and how do we construct such codes. So, these are some of the aspects that we look at in the subsequent modules.

Thank you very much.