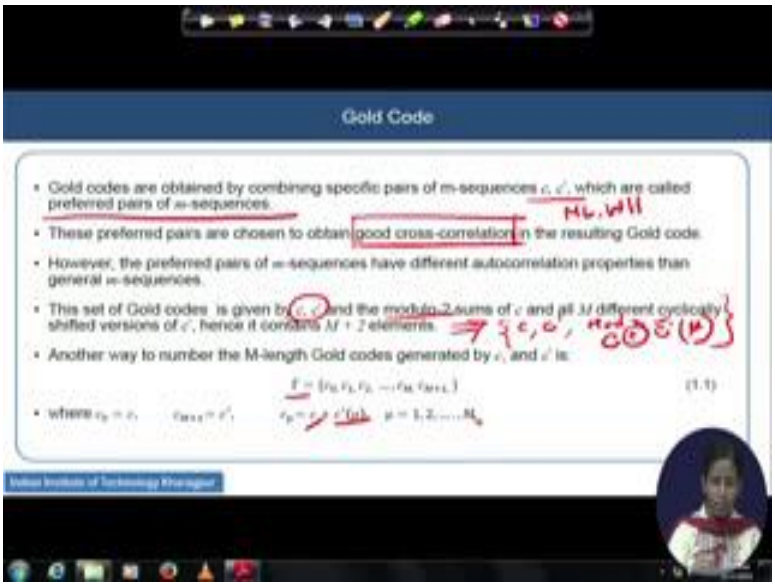
Spread Spectrum Communications and Jamming
Prof. Debarati Sen
G S Sanyal School of Telecommunications
Indian Institute of Technology, Kharagpur

Lecture - 16
Generation Mechanism and Properties of Gold and Kasami Codes

(Refer Slide Time: 00:47)



Hello students, we were discussing different kind of the generation mechanism of different kind of the codes that are utilized for the spread spectrum communication systems. Today the continuation of our last two module, we will today we will discuss about the gold and the kasami codes their generation process and the characteristics. Remember we have already discuss about the choice of the code will be based on autocorrelation and the cross-correlation property of the codes because we have understood that either the codes will be utilized for the specific delay estimation or specific localization positioning where the autocorrelation peak is of prime importance. And otherwise, it will be utilized for separating the multiple users in a code division multiple access kind of the scenario. Where cross-correlation property between the codes will be of very important property that should be that will govern the choice of the typical kind of a code.

So, we are really looking for the codes who are having the very low low-cross-correlation properties because in the wireless communication scenario multipath delays cannot be avoided. So, two codes will never be synchronized and only the orthogonality between the code set is not sufficient to prevent the interference between the code sets. And now we are talking about the kind of the code gold code, we have with the understanding that it will help us to realize very very better actually autocross-correlation properties. Gold codes we obtained by combining the specific pair of the m-sequences.

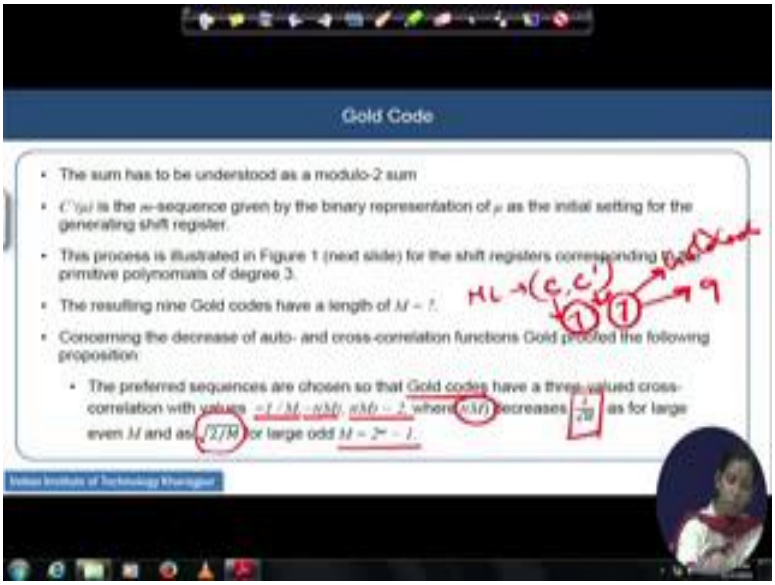
Let us start with the pair of the m-sequence c and c prime. Remember the pair of the m-sequence or not all the m-sequence can help you to generate the gold code. The typical pair of the m-sequence that will be helpful for generating the gold sequences are called always the preferred pairs of the m-sequence. This preferred pairs of the m-sequence are chosen such a way that it finally leads to a very good cross-correlation in the resulting gold codes, as a target is always the cross-correlation reducing the cross-correlation further than the code that we have already discussed. For example, if you are starting with ML sequencing, so we are demanding the cross-correlation property to be going down to compared to ML, compared to Walsh Hadamard correct.

So, let us see whether gold can give us that or not. The preferred pairs of this m-sequence they have the different autocorrelation properties than the generated gold sequence definitely it should be. The set will be generated like this. Given this pair of c and c prime, we will go ahead with a modulo-2 operation of c with a capital M circularly shifted version of c prime. And hence the generated gold code will be consisting of c, c prime and mod 2 operation of c with circularly shift in version of c prime. So, I am writing it a c delta. So, this circular shifted version of c prime the way you are shifting it is the mu.

So, if the capital M numbers of the; if you start with the length of capital M and then it will be generated of M length gold code the way you will be generating that M length gold code you will be having basically actually 7 M plus 2 number of the codes available inside that set. And if you include the original two preferred pairs inside that gold code. So, we can write in another way the generated gold code is written like; this is generation mechanism we will discuss later in the next slide.

Let us understand now that if tau is a generated gold code and then it will be written like this, c 0 is the first codes; c 1 is the second, c 2 like that you will be getting c m plus 1 code. Where c 0 is basically the one of the pair code chosen from ML sequence, c n plus 1 is another pair code chosen from m ML sequence. And each of these codes are generated by modulo-2 bit by bit modulo-2 operation of codes c with the circularly shifted version of c prime; this circular shifting can go from 1 to capital M.

(Refer Slide Time: 05:18)

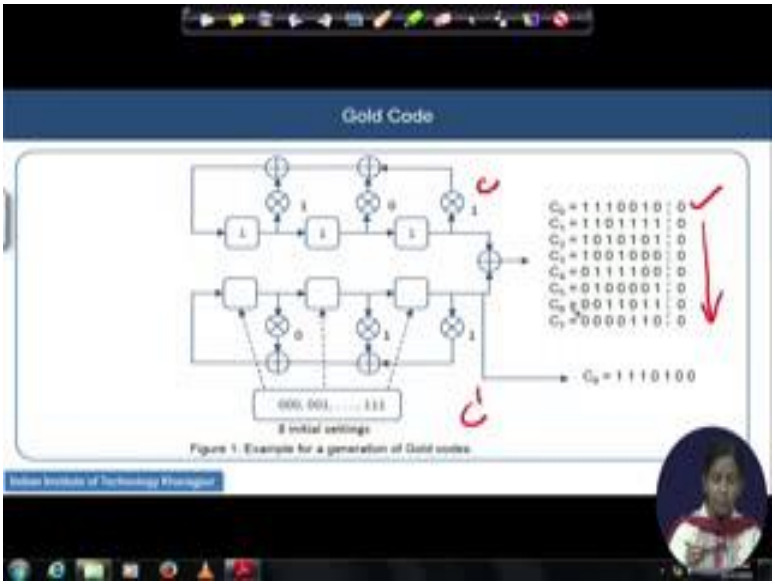


So, we will see in the figure one in the next slide the way actually the code is actually generated in the circuit. And we will start with a capital M value length of the value is equal to 7. So, we have chosen ML sequence, we have chosen a ML sequence and the pair c and c prime and both of them are having the same length had the length chosen we are starting with this with 7 also. And the resulting the code number of the code that we will be ending up with is equal to 9. So, the length of the code which we started is equal to 2 is length is equal to 7 we started with a pair of the ML sequence code; the length of the gold code will be also 7, but such seven length code we will get equal to number 9. So, nine codes gold codes we will be generating; each of them are having a length of 7. And the paired codes who are helping us paired codes of the ML sequence who are helping us to generate this code they are also having a length of equal to 7.

Concerning the decrease of the auto-and the cross-correlation function of the gold prove the following prepositions. That the preferred sequences that are chosen to generate the gold codes they will have a three values cross-correlation function. Such new chose the sequence such a way that will end up with a gold code and that gold code will show you the three stages of the cross-correlation values. They are like this 1 will be minus 1 by M, another will be minus t of this function M, the third will be t of function M minus 2. But this t M he will decrease at the rate of two by square root of M for the large value of even value of capital M. And if M is equal to large odd given by 2 to the power small m minus 1 then the decrement of this t M will be given by square root of 2 by M. So, the values with which you are ending up the cross-correlation values they are minus 1 by M, minus t of M and t of M minus 2. So, this depending on whether the value of the capital M - the length you are choosing is even or odd, large even or large odd, you will be getting actually three different state of the autocorrelation cross-correlation values.

So, this cross-correlation function that we are finally ending up with we can see that they are for below than the cross-correlation values of the Walsh Hadamard code that we have earlier discussed and as well as the ML sequence codes that we have already discussed. So, with respect to the cross-correlation values, gold codes are really very good. Even if you see the autocorrelation value of this gold code, the autocorrelation value is also very nice it is though actually the rate at which the autocorrelation decreases this is really high compared to the Walsh code, but still the value of the autocorrelation peak is much higher than the Walsh Hadamard code.

(Refer Slide Time: 08:42)



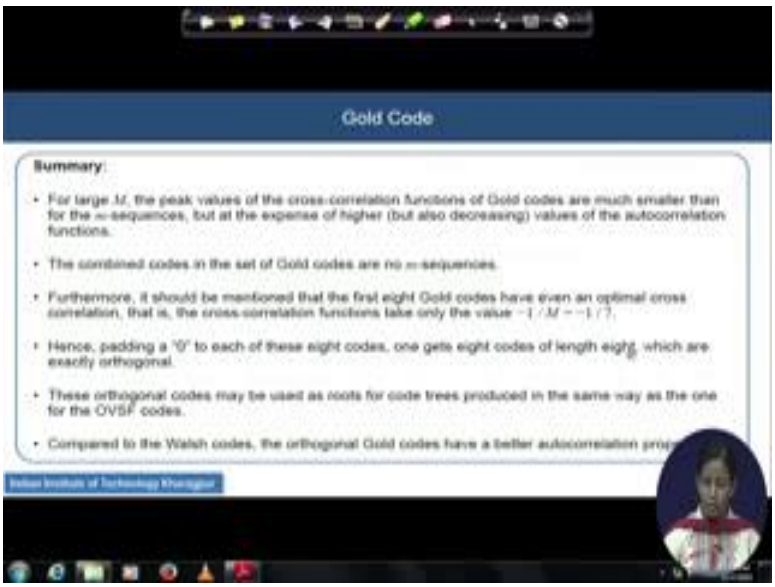
So, the way we will generate the gold code is like this. So, there are two different ML sequence generation process who is supposed generating the bridge set the two structure of the LFSR, who is generating the ML sequences. And the ML sequences are here we are denoting those ML sequences as c and c prime. And the final goal code will be generated by the more bit by modulo-2 operation of this c and circularly shifted version of this c prime. The circularly shifted version we can easily create by changing the initial situation. So, we will be keeping on changing the initial situation initialization of this LFSR I mean all the memory stages M-stage values. And for each and every stages the bit that will be generated here will be giving 1 1 kind of the new codes associate with it.

Say for if I considered that 0 0 C H code is generated for 0 0 0 situation, then once you are loading it will be not getting it is an XOR operation going on. So, when once all the they must 0. So, you will be continuously getting all the values that are stored here coming out and he will be generating that you will be generating the first C 0 code. For C H actually we will be taking the output from here directly, because that is the direct value C 1 that is of direct value C prime.

So, one value is that only one value is the C 0 original one, another will be the start value of the C 8, C 8 is the only original value of the another pair of the ML code. And

remaining part all the C 1 to C 7 you are generating bit-by-bit multiplication of this C 0 with that circularly shifted version of my C 8. So, you can check it also suppose if I am circulate if I circulate it will be 0 in the first set, 0 or 1 will be giving me the 1. So, next is 0 or 1 again will be giving me the 1 that and then 1 with XOR with 1 will be giving me 0. So, like that the C 1 if generating for C 2 to generate, we have to circular shift it once more and then you map and try to do the bit by bit XOR operation you will see that C 2 is generated. Please check at all the generation process of all the codes and check whether you can generate all the codes I have reported here to understand the way the gold code is generated.

(Refer Slide Time: 11:30)



We were discussing actually the few properties of the gold code, and we have understood that gold code can give us very nice cross-correlation properties and the value of the cross-correlation peaks are very, very low. So, see whenever you are getting the cross-correlation peaks, if its value is really low compared to the close to the noise level then also actually you can it is good actually because you can compare the interference close to the noise level. And it can be discarded, and it can have a minimal effect on the detection process of the original data.

So, in that sense actually even if we can select a code in a multiuser scenario where the cross-correlation properties are very close very, but is boiling down towards a very low value and we are giving very low value of the cross-correlation peaks then this kind of the code is always preferable for enmity is a kind of scenario and good autocorrelation property as I understand for my own understanding for my own detection for my intended kind of the reception, I need a very high very good autocorrelation for peaks.

So, gold is a only code who can do justice for the both. I mean who can give you reasonably good amount of the autocorrelation peaks and reasonably less amount of the cross-correlation peaks that is why the name is gold. This is the only code available where you can get a very good cross-correlation both cross-correlation and autocorrelation both you can give a very nice values, which is very rare. Usually we can get any other codes if you whatever we I have shown in the series, you will get either cross-correlation value very nice or the autocorrelation value very nice; gold is the only code where both the justice is both is done.

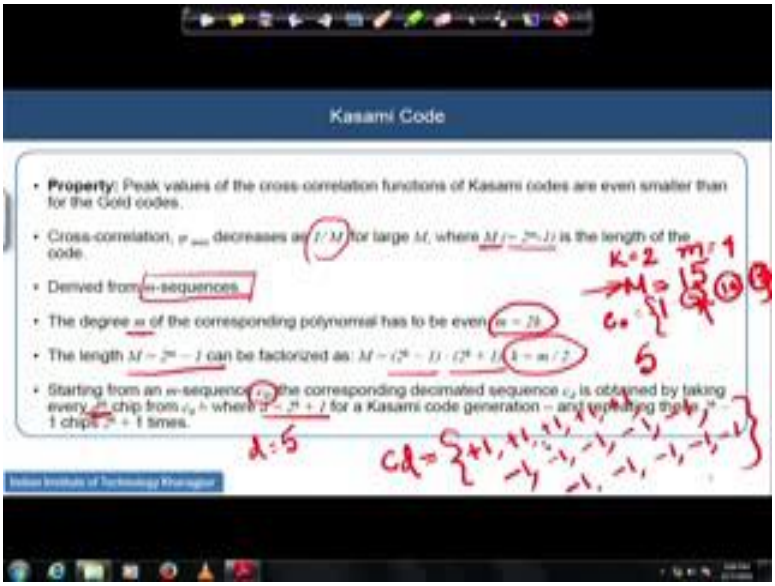
And as though we are generating the gold code from two m-sequence code because C and C prime both whatever we have started with both are them ML codes. But remember after combination the ML properties this there are no more the C the way the gold code is generated after the generation of the gold codes, they are no more actually are m-sequences. So, you can never see actually the property of m-sequence aside the gold code in that sense, the combine code in a set. And it should be mentioned that the first set of the gold codes that we have seen in the earlier slide, they will have the cross-correlation value the first eight gold codes that we have a even and optimal cross-correlation. And for all those eight gold codes, he will have always the value cross-correlation value is equal to minus 1 by M which is minus 1 by 7 is it can be proved actually.

And hence if I add a 0 to each of them if pad 0 to each of them. So, very nicely you will get if I pad 0 to each of them I will get eight-eight length codes. So, it is so nice actually and all of them will show you the orthogonality. So, this is generated first the seven length codes and as the first eight codes generated like this way, first eight codes and you know the the ninth one always show the cross-correlation property is equal to minus 1 by

capital M, here it will be minus 1 by 9. So, if I pad zeros to each of them they are also banality prevails and the good part is that now you are getting the eight different codes of length 8.

So, now, see in the OVSF we have seen that in order to vary the data rate, you need the different amount of the spreading factor for the codes, where from you will generate the child codes. So, you can actually generate you cannot pick up the seventh length code gold code as a mother of this OVSF code or eight length code to generate the separate set of the OVSF code. So, gold code can gold codes can be nicely utilized as a mother or the parents of generation of the OVSF codes in that sense, because generating multiple set of the orthogonal codes by simply adding the zeros that nice property no other code will give you.

(Refer Slide Time: 16:14)



Compared to the Walsh Hadamard code we will see that they are having the for better autocorrelation property. So, they are the gold codes are better in terms of autocorrelation from ML as well as your Walsh Hadamard. They are better in the cross-correlation property compared to the Walsh Hadamard as well as the ML. And with that gold codes before comparing the gold codes with all of them the number of the codes that we can

generate the autocorrelation, cross-correlation properties and their availabilities and all, let us finish of the series first the last family is the kasami code.

Kasami code even though gold code could improve the cross-correlation peak values compared to the ML and Walsh Hadamard people, try to further lower the cross-correlation property in that such process where the design the kasami code where the cross-correlation function would be lowered even down compare to the gold codes. Here the cross-correlation decreases as 1 by M for large value of the M, where this a capital M is given by 2 to the power small m minus 1; and this is the length of the kasami code also; and this are again derived from the m-sequence. So, ML sequence or m-sequences are the mother of all the other code generation. And this degree m it is always corresponds to have be always to be even number, we will take an example and see how the kasami code is constructed.

So, if small m is equal 2 to the power k. So, this guy will be 2 to the power small m minus 1 given by small m minus 1. And then hence 2 to the power 2 k minus 1 can be further constructed it can be a further actually factorize by to the power minus 1 into 2 to the power k plus 1, where my k value is equal to m by 2. So, the way the sequence will be generated is like this. You start with the m-sequence C 0, and then correspondly decimate the sequence C d you generate the decimate sequence C d by taking the every dth chip from this C 0, where your d will be given as 2 to the power k plus 1 for kasami code generation. And repeat this 2 to the power k minus 1 chips for 2 to the power k plus 1 times. It is hard to understand. So, let us take an example.

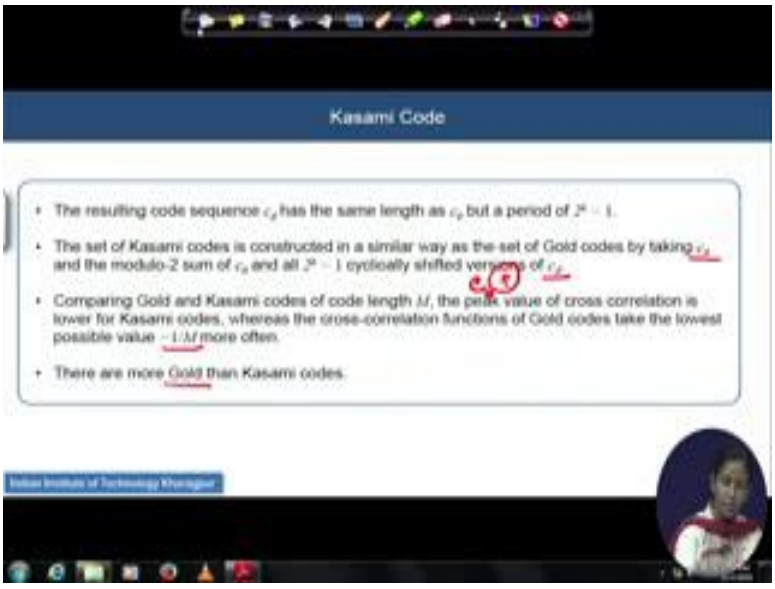
Suppose k is equal to 2. So, the value of small m is equal to 2 into k which is equal to 4. Now, the length capital M is supposed to be 2 to the power 4 minus 1 is equal to 15. So, I have started with the m-sequence of length 15 also. So, suppose C 0 is having 15 chips or 15 bits inside and where actually the number I am going, so number is plus and minus only, but I am having 15 search values one to 15 search values I am having. And then other way I am choosing the dth chip is such that d is equal to 2 to the power k plus 1, if my k value is equal to 2 then the value of d will be 5. So, every feed chip you pick up. So, chip number 5, chip number 10, chip number 15, this we will be picked up. And once they are picking picked up then you have the kasami code C d will be generated by

repeating each of this chip, how many times 2 to the power k plus 1 time. So, 2 to the power k plus 1 is again value is equal 5. So, I have peach the fifth chip and then I will repeat the fifth chip five times. If the fifth chip value is equal to say plus 1, so I will repeat the fifth chip five times.

Then I will pick up the tenth chip because d value is equal to 5. So, first the value will be picked up is a fifth chip then you pick up the tenth chip. And if the tenth chip value is equal to minus 1, so that value will be repeated five times. Suppose the last value is the 15th, so then again you pick up it is again supposed to be minus 1, then I will repeat again this value five times. So, this is my generated kasami code finally. So, this is the way I generating the code, but remember whatever the how many number of the chips d value is we will be able to get is always 2 to the power k minus 1 chips like this you will get and each of this chip will be repeated first five times to generate your kasami code.

So, the length of the code will be equal to your length of the sequence that length of the ML sequence or m-sequence with which you started because C here also you are having the length equal to 15. But its period will be only 3, I mean the 2 to the power k minus 1 it will be the period by which actually you are which actually the code will be repeated.

(Refer Slide Time: 21:43)

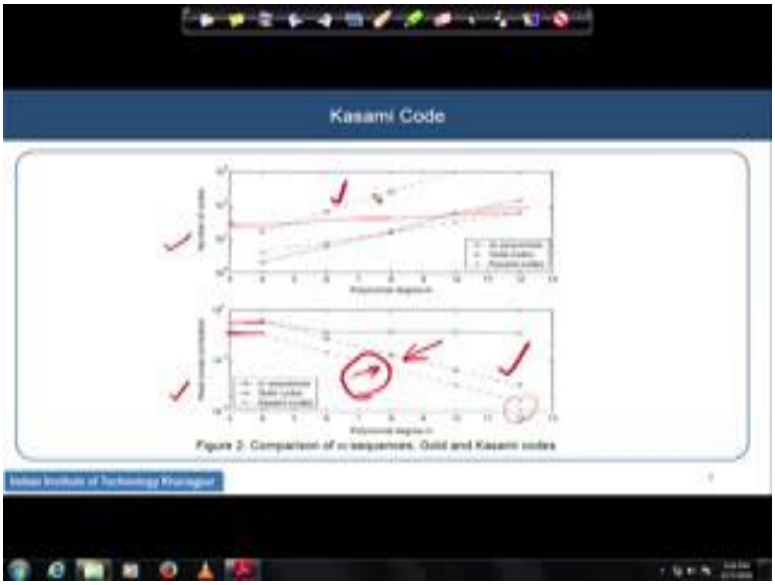


And the set of the kasami code is also constructed using the same kind of the circuit design that we have shown for the gold code generation, and there you will be generating the C 0, C 0 is a your own code. So, ML sequence from where you are starting. And you do the modulo-2 sum of the C 0, and C 0 all the modulo-2 sum of the C 0 all to the power k minus 1 cycle shifted version of this C d. So, C 0 and the 2 to the power a minus 1 cyclically shifted value of the C d, so you first construct C d from C 0 the way explain. And then you take the cycle cyclically shifted version of this C d, and they you do the bit by bit modulo-2 operation with your C 0 to generate the kasami code.

And that is the way I can generate a set of the kasami code last slide talked about the generation of one such kasami code. And if I wish to have a set of such kind of the kasami codes, so there you have to move the C d by cyclic shifted way and then keep on doing the modulo-2 operation with your C 0 to generate the set.

So, if I compare now the gold code and the kasami code cross-correlation values, you will see that the peak values of the cross-correlation is for far lower than the kasami code. Whereas the cross-correlation function of the gold code takes the lowest possible value is equal to minus 1 by M most of the time; here it is not that. And moreover if I compare the number of the codes available in the gold as well as in the kasami, we will find that the gold codes are much more available compared to the kasami codes.

(Refer Slide Time: 23:37)



We will see a nice example and the comparison in the next slide, where we will see that compare the number of the codes available in between m-sequence gold code and kasami code and their most important property where we are interested in its peak cross-correlation value and where we are standing. Let us first start the comparison over the number of the codes available. So, if I am increasing the polynomial degree small m and then the I will check the number of the codes available. So, if I increase it, remember the m-sequence codes are easily available. And they go with the increase of the polynomial degree of small m, you can reach approximately with the polynomial value of 12, you can reach up to say 8 into 10 to the power 80 around 80 number of the codes you will be able to generate, for small m equal to 12.

Whereas, for small m is equal to 10, you will be able to generate far larger number of the codes already more than 100 number of the codes almost close to the 1000 we are here. So, with 10 value of the order or we can generate around 1000 gold codes. So, if I check the availability is a gold codes are no other codes are even compatible in terms of the number of the codes available is compared to gold. Kasami actually initially for lower number of the orders kasami codes are available, it was the kasami codes or more, but if you are crossing the degree h, in the kasami codes are then the availability is heavily decreasing compared to my m-sequence. So, in terms of the number of the codes

available, gold is the winner; and ML sequence is loser in terms of the when the polynomial degree is increasing.

Let us come to the peak cross-correlation comparison. Again the comparison will be between in between three codes m-sequence, gold and the kasami. See, for the m-sequence, we understand that it understand it bit decrease the cross-correlation peak value decreases, when the polynomial degree is up to the 6 and then it becomes a constant value always, and even if you are increasing the number of the orders. So, here actually if I compare the both, so m-sequence with the higher order only the number of the available of the code little bit increases, but beyond 6 actually availability also decreases and the peak cross-correlation value is also becoming constant. There is no further improvement, which is not a favorable situation and satisfactory situation for a multiuser environment to choose this code for this time for CDMA kind of application.

And hence come to the gold codes, gold is showing us continuous decrement of the cross-correlation value with the increment of the polynomial degree as expected, because it has a three different set of the cross-correlation peak values. But, see the kasami, kasami is even lower and always actually with increase of the polynomial degree there is a gap of around here, it will be approximately 7 or 8 into 10 to the power minus 1. So, and from there, it is the about peak values is like that and here we are having several times lower at the same polynomial degree and that gap is continuing. It is continuing with the increase of the degree.

So, the peak cross-correlation value in terms of that if I compare, so peak cross-correlation is the best in terms of in the kasami codes. But, if I check the availability of the kasami code, so with the increment of the 12 order number 12, you will heavy decrement in the cross-correlation value is possible. But at the same time, the number of the codes that is available hardly will be your 2 into 20 or 30 like that it will be hardly 20, 25number of the codes available.

So, the number if I am heavily interested to give address or multiuser scenario once again I am coming back where actually this very good cross-correlation property is required. So, for such scenario though the cross-correlation property has heavily

decreased, but there is because of the availability of the sufficient number of the kasami codes for a dense network, we cannot actually provide this code for the multi user environment. So, finally, again we are even we understand that the kasami is superior than gold, gold is not having that much good cross-correlation property, but finally, gold wins the situation because the number of the codes available here is much more.

So, in practical environment where the den set work needs to be designed gold will be the first choice because of the large number of availability of the code and very pretty good and pretty reasonable amount of the low cross-correlation values available. But if you are density is not that much that much higher, and you have very few number of the users turned on at a particular time, then you can utilize also the kasami code, but that will be a choice it is a variable choice depending upon your system requirement. If you are a cross-correlation values are really very important to go ahead with then you do a compromise between the choice of the kasami codes and not supporting large number of the users.

But if large number of users is the first choice of the support, then you please go ahead with the gold code without compromise of the larger amount of the interference coming from the in between the code, and handle that interference it is the code interference. So, inter code interference you handle that interference by some other means by some other signal processing algorithms in the receiver.