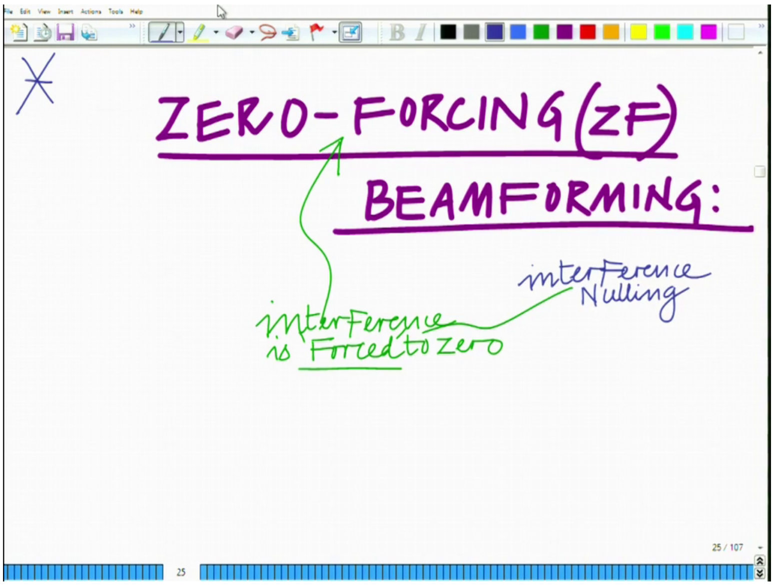
Applied Optimization for Wireless, Machine Learning, Big Data
Prof. Aditya K. Jagannatham
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

Lecture – 37
Practical Application: Zero-Forcing (ZF) Beamforming with Interfering User

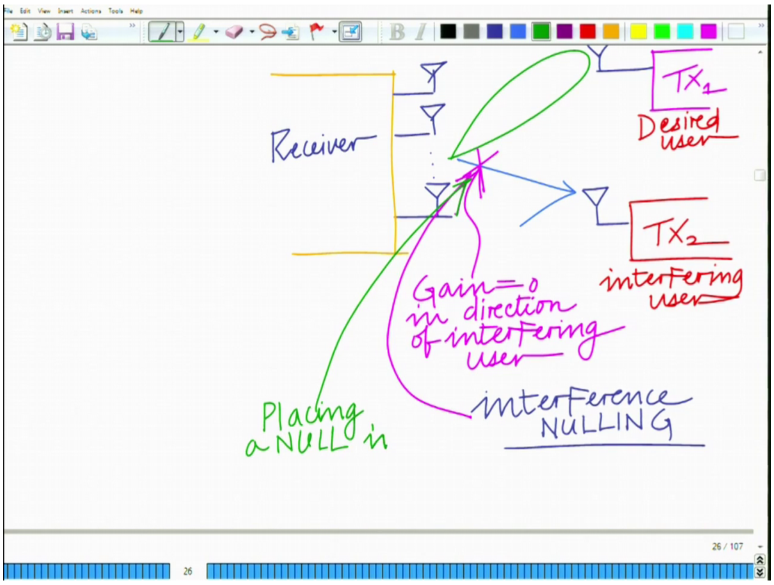
Hello. Welcome to another module in this massive open online course. So, you are looking at canary optimization and practical applications of various optimization problems in the context of Beamforming, alright. We have looked at different kinds of beam forming, the beam forming, the original beam forming problem, we have also seen beam forming with interference. In this module let us look at yet another kind of beam forming; that is, beam which is known as Zero-Forcing beam for me ok.

(Refer Slide Time: 00:43)



So, what we want to look at is a different kind of beam forming which is termed as zero-forcing, beamforming, you can also call this as ZF- ZF for zero-forcing. Now what happens in zero-forcing beamforming? In zero-forcing beamforming the interference is null, interference is made 0. So, we are forcing interference to 0, that is why is forced is interference is Forced to 0, that is why this is known as zero-forcing beam following; is also known as interference nulling. You can call this also as nulling the interference, interference is null, interference Nulling.

(Refer Slide Time: 02:02)

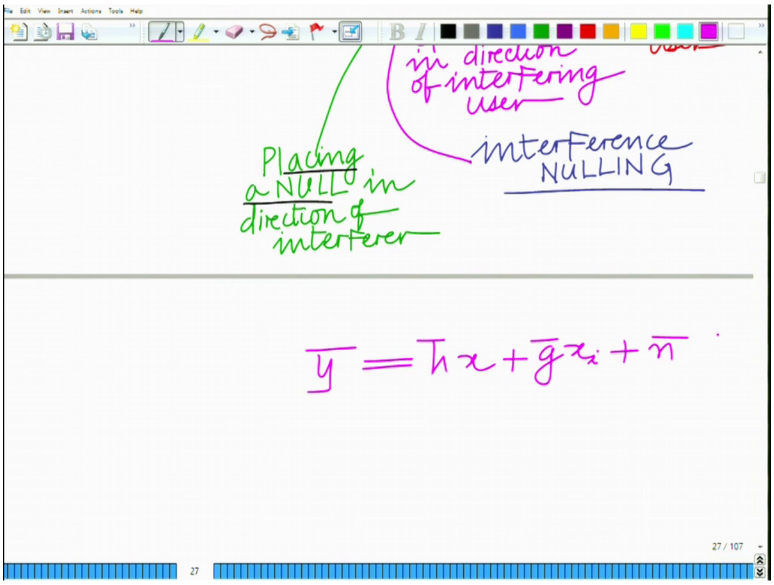


What happens in zero-forcing beamforming? Well have seen beamforming where you have a multiple antenna array, and you have these multiple antennas, and let us see. So, this is a receiver, and you have the desired user, and you have your interfering user this is your interfering user.

This is the desired user, and what you do in this is that you again maximize the signal gain in this direction; direction of the desired user. While in the direction of the interfering user, in the direction of the interfering user, what you do is, you simply place a null; that is, you make the signal equal to 0 ok. So, equal to 0, or gain equal to 0, you can say signal gain in the direction of interfering user. Gain equal to 0 in the direction of the interferences, or you can say the interference is nulled. Therefore, you are nulling the interference.

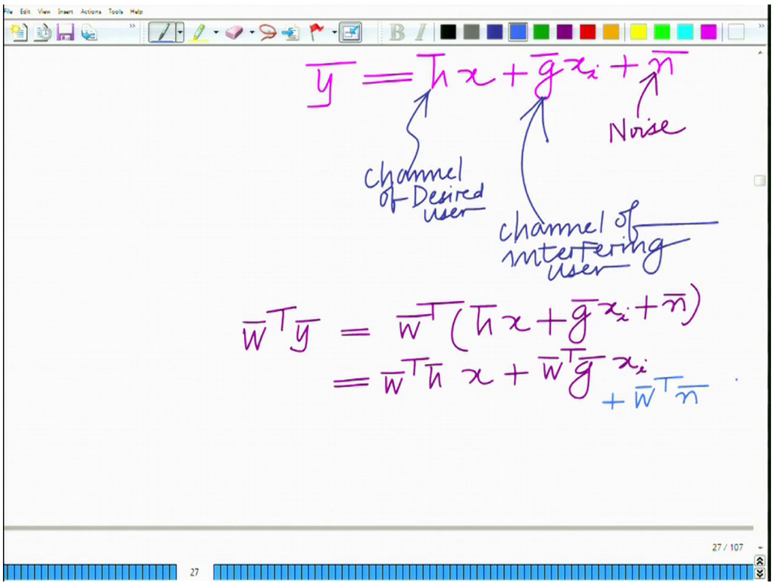
This is termed as interference nulling, your nulling the interference by basically ensuring that the gain in the gain in the direction of the interfering user is 0. This also termed as interference nulling. Or this also in fact, termed as placing a null in the direction of interfere. So, this can also be thought of as placing a null. Already this various no mental kinds of nomenclature, this is Placing a NULL in direction of interfere; also termed as Placing a NULL, you are placing a null in the direction of interfere.

(Refer Slide Time: 04:59)



Let us look at the procedure to do this. Let us again go back to a system model that is y bar equals h bar x plus g bar x i plus n bar this model is similar to the model that you might remember we had seen before. That is, this model y bar equals h bar x plus g bar x i plus n bar; where you can see once again.

(Refer Slide Time: 05:51)

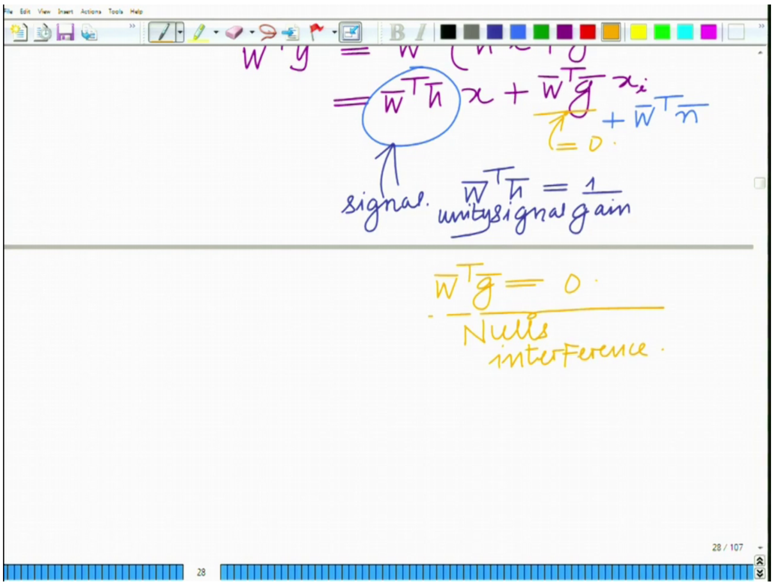


This is basically the channel vector of the desired user, the channel of the desired user. This is the channel of the interfering user. This is the channel of the interfering user ok.

And you place w bar is the beam forming vector and w bar. So, you are performing beam forming correct similar to what you have done before.

So, we have w bar transpose y bar equals w bar transpose x bar x plus g bar x i plus n bar; where n bar is the noise this is the additive white Gaussian noise as usual. So, this is your w bar transpose h bar x, plus w bar transpose g bar into x i, plus again the noise output that is w bar transpose n bar.

(Refer Slide Time: 07:25)

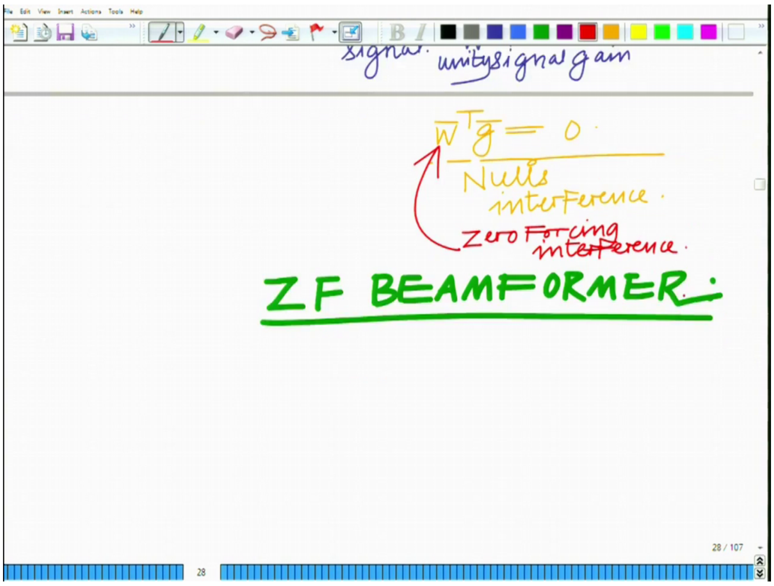


Now, this is the signal part, and you ensure that signal gain equals 1, the gain in the direction you can also term this as the gain in the direction of the desired user. So, this ensures that signal gain this ensures basically your unit is signal gain.

Now here what we do is to null the interference we set this interference term to 0. So, what we are doing is we are setting is w bar transpose g bar equal to 0. So, this basically nulls this basically nulls the interference. So, by setting w bar transpose g bar equals 0 what you are ensuring is that, you are placing a null in the direction of the interferer or basically you are not able to receive or basically you are you are effectively suppressing, or you are effectively not just suppressing, you are effectively zeroing whatever is the signal that is received from the interfere. So, that is basically ensured by this condition that we were transpose g bar equal c.

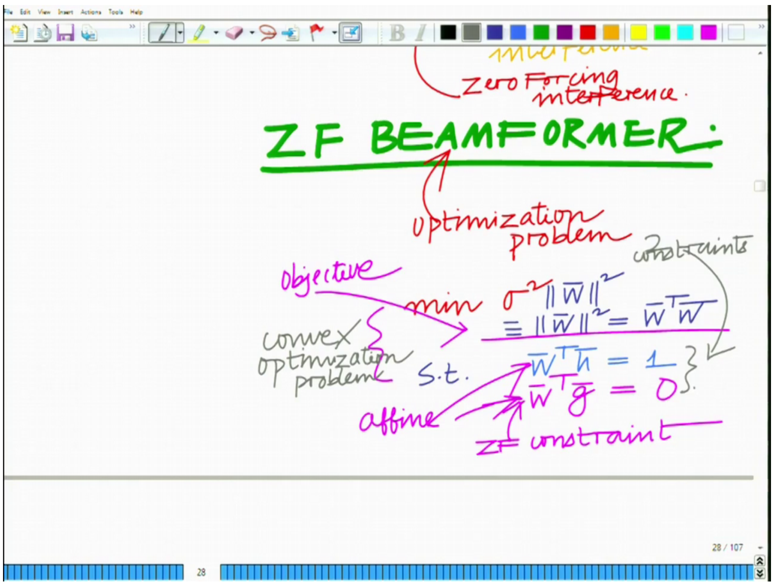
And therefore, now your optimization problem the resulting zero-forcing beam former, optimization problem for the ZF beam former.

(Refer Slide Time: 09:04)



And by the way this is your zero-forcing condition in case you are wondering what is 0 forcing. So, this is basically you are forcing the interference to 0. You are forcing the interference to 0.

(Refer Slide Time: 09:44)



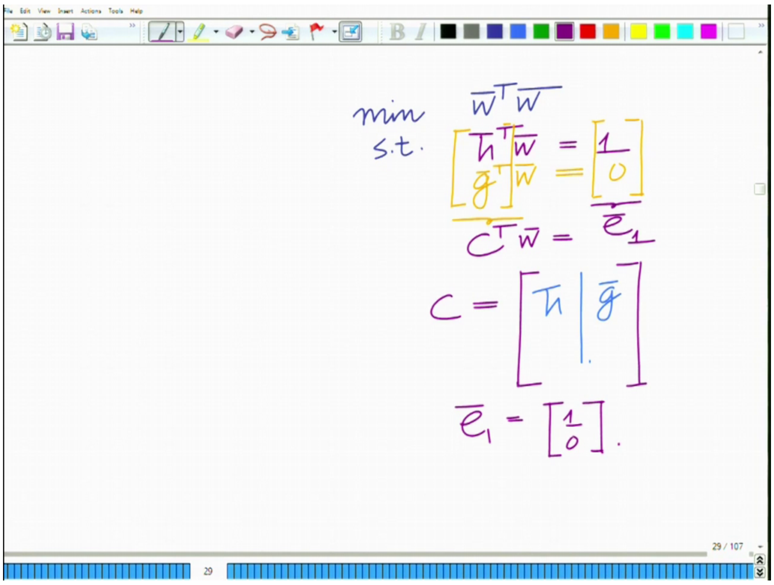
And therefore, the resulting optimization problem for a zero-forcing beamforming, you minimize the noise power as usual that is sigma square norm w bar square; which we said is equivalent to minimizing norm w bar square, because sigma square is constant, which is basically nothing but w bar transpose w bar. So, you minimize w bar transpose w bar. Now subject to the constraint, now you have 2 constraints. In fact, previously you had only one constraint. So, you have w bar transpose h bar equals 1 unit again in the direction of the signal.

Now, you have another constraint, that is w bar transpose g bar equals 0, this is your ZF constraint. This is your ZF constraint: this is your objective function, which is again convex. This is an affine constraint, this is also an affine constraint, both of them are linear. So, this is basically now your convex optimization problem. This is again a convex optimization problem, similar to what the objective is convex constraints are in fact, affine that linear constraints of convex.

This is a convex, this is a convex optimization problem; however, now you see that you have 2 constraints, correct? Unlike the previous one where you had only a single constraint, you have to interact in a general optimization problem you can have multiple constraint, not just one constraint. That was a very I mean previous problems were very simple rather simple. So now, you have 2 constraint and you can in fact have multiple constraints. In fact, in this scenario itself you can see that if you have more than one interfering user. So, if you have let us consider a schematic; where you have more than one interfering user.

So, you have t x 3 and again you want to place a NULL along the direction of t x 3. So, depending on the number of interfering users, you can see in this scenario you have constraints. In fact, if k is the number of interfering users, you have k plus 1 constraint. One is the signal gain that is unity signal gain, plus k null constraints for the k interfering users only. So, the number of constraints grows with the number of interfering users.

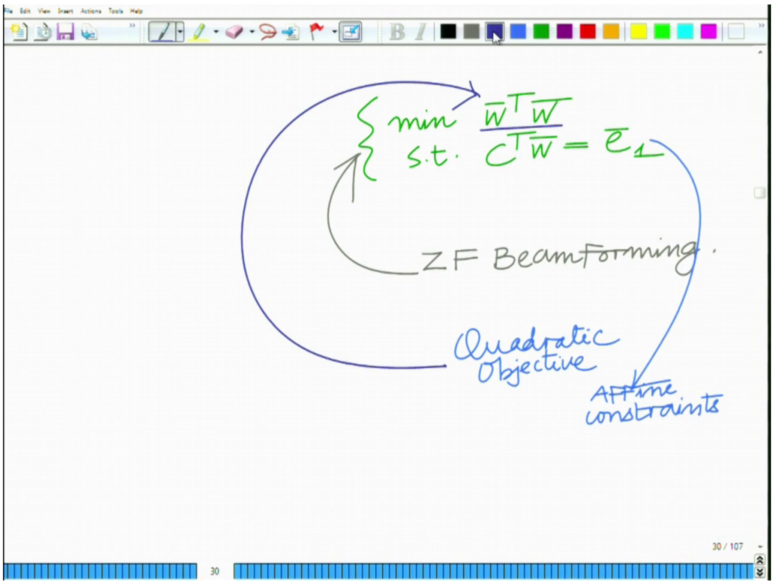
(Refer Slide Time: 12:36)



And therefore, now this optimization problem; again I can write this as w bar transpose w bar subject to the constraint. Well, I can write this as w bar transpose h bar is h bar transpose w bar. So, I can write this as subject to the constraint, h bar transpose w bar equals 1. And g bar transpose w bar equals 0.

And now I can make this as a matrix, this as a vector, I can call this matrix as c transpose. So, c transpose w bar equals this vector e bar 1; where c is the matrix you can see this is the matrix which is this matrix. It is first column will be h bar and the second column will be g bar. So, this is the matrix and e bar is this vector, e 1 bar is this vector which has one in the first position and 0.

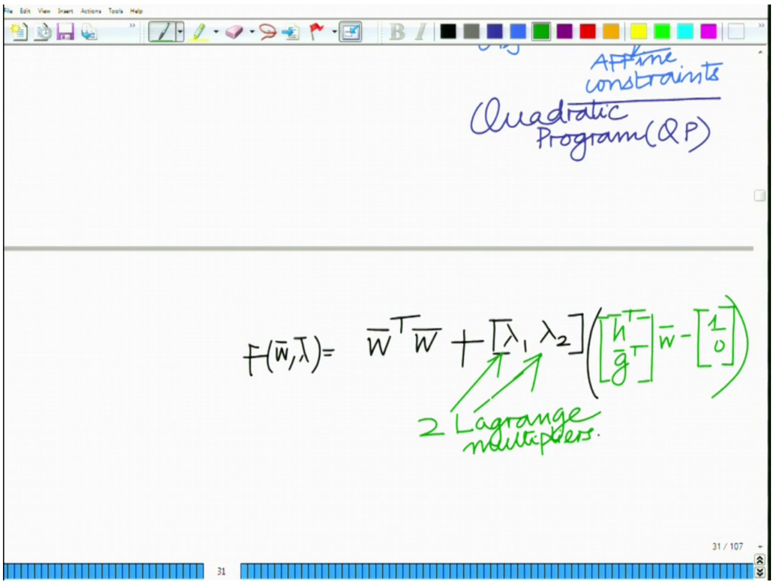
(Refer Slide Time: 14:06)



So I can write this optimization problem as minimize known w w bar transpose w bar subject to c transpose w bar equals e bar 1. And this is basically the optimization problem now for my zero-forcing Beam Forning. This is the optimization problem for zero-forcing Beam Forning.

In fact, this is known as a quadratic program. See what you have here is; you have a quadratic constraint, a quadratic objective function. So, you have you have a quadratic objective function, quadratic objective function. So, this quadratic objective function and linear constraints, affine constraints or rather we put these things as affine constraint.

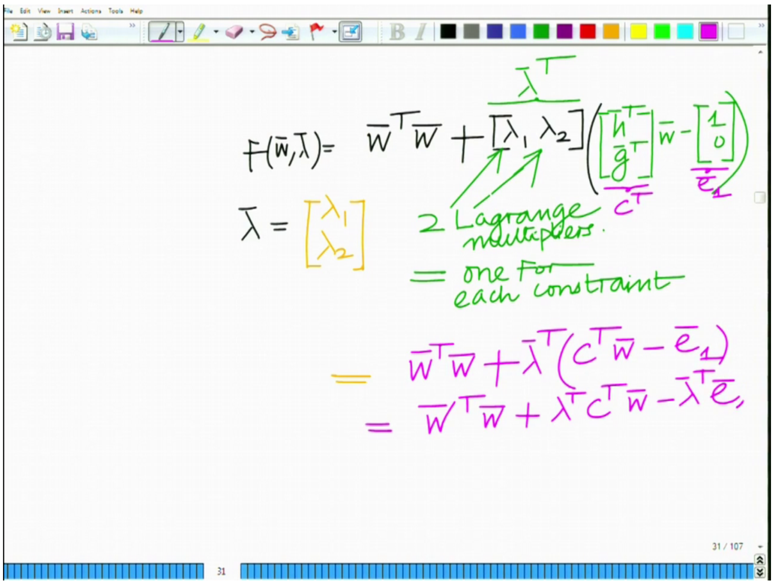
(Refer Slide Time: 15:41)



And therefore, this is basically termed as a quadratic program. This is therefore, termed as a quadratic program, this kind of this type of an optimization problem is termed as a quadratic program. And now, what we want to do is; we want to solve this quadratic program to obtain the zero-forcing being former.

So, I form the Lagrangian which is f of w bar from our lambda bar, we will see that it will be function of a vector lambda bar; which is w bar transpose w bar plus, now you see there are 2 constraints. So, I will need one Lagrnage multiplier for each constraint. So, lambda 1 lambda 2 into c transpose; that is basically your h bar transpose g bar transpose times w bar minus e bar a 1 bar ok. So, this is your first constraint, h bar transpose w bar minus 1 g bar transpose w bar equals 0. So, you have 2 constraints so, 2 Lagrnage multipliers, one for each constraint.

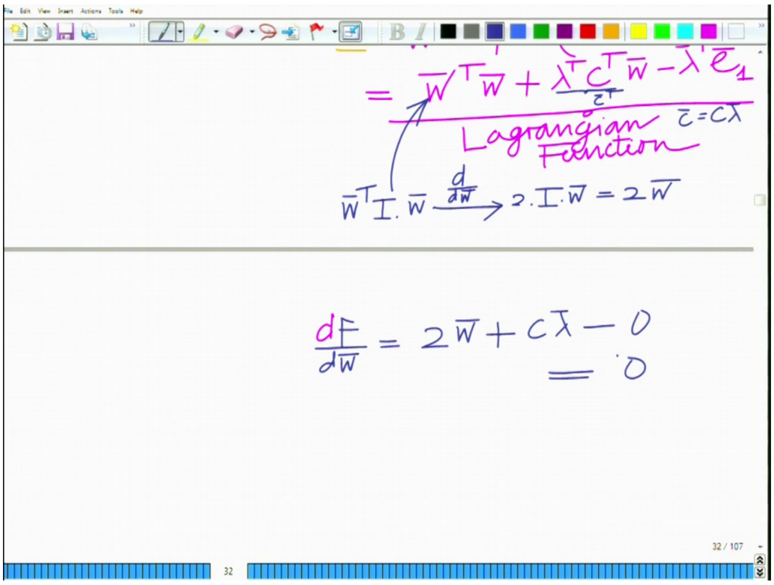
(Refer Slide Time: 17:23)



So, you have to recognize multipliers equals 1 for each constraint. And therefore, this is the rho vector lambda bar transpose. I can write this as lambda bar transpose; where lambda bar is a vector containing the 2 Lagrnage multipliers, that is your lambda 1 and lambda 2.

And therefore, this will be w bar transpose w bar plus lambda bar transpose times this is your c transpose, this is your e bar 1. So, this is your c transpose w bar minus e bar 1 plus equals which is equal to w bar transpose w bar plus lambda bar transpose c transpose w bar minus lambda bar transpose e bar one that is your Lagrangian.

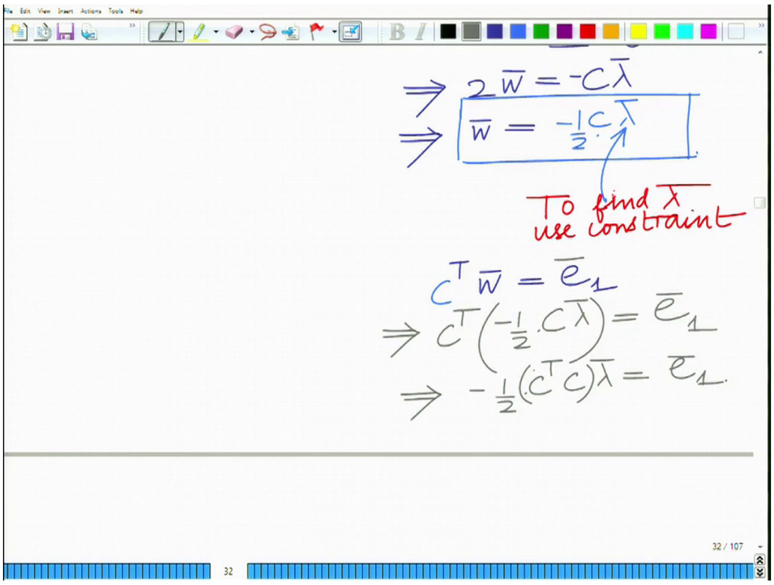
(Refer Slide Time: 18:35)



This is your Lagrangian function. Now we are going to differentiate this, right compute the gradient of this Lagrangian function with respect to v bar differentiate this with respect to v bar, I am sorry w bar. So, I compute derivative of w bar the transpose w bar with respect to w bar that is twice w bar. You can also think of this as w bar transpose w bar is w bar transpose, identity times w bar; where the derivative is so, if you differentiate this with respect to w bar, you get twice identity into w 1 which is nothing but twice of w bar.

So derivative of w transpose w bar is twice w bar. And in any case you can see w bar transpose w bar is w 1 square plus w 2 square. So, on up to w l square if you differentiate it with respect to each w I you have 2 w i. So, that is nothing but the vector 2 w bar. And plus lambda bar so, this is of the form lambda bar c transpose. So, this is of the form your c bar transpose w bar where c bar equals c into lambda bar. So, the derivative is simply c bar c bar transpose w bar the derivative with respect to w bar is c bar. So, this is c lambda bar, derivative of lambda bar transpose e bar w bar with respect to w bar is 0. So, simply that is 0 so, minus 0 and this we set it equal to 0 ok, the gradient is being set equal to 0 like for any optimization problem.

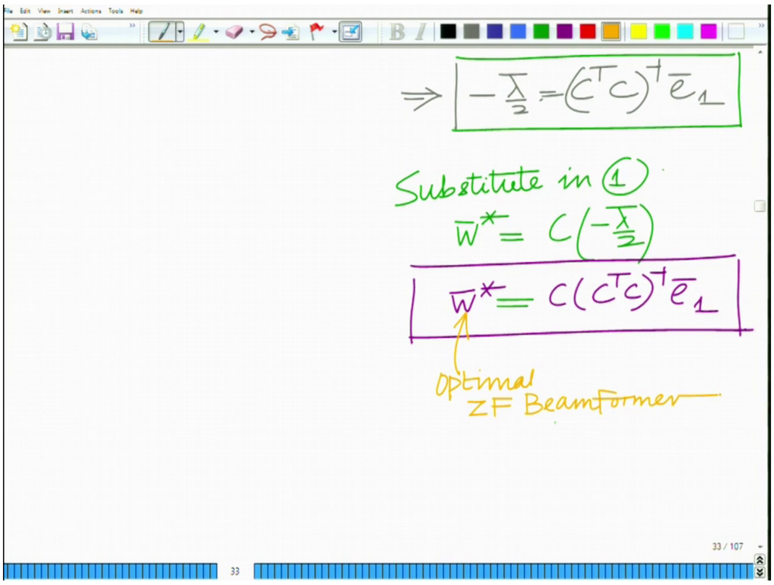
(Refer Slide Time: 20:31)



Right, you do for any optimization problem to find the extrema. And this implies now, therefore, what you have now this implies that twice w bar equals c times minus c times. Now here note that you cannot interchange the c and lambda bar, because previously lambda was a scalar. So, you can simply bring it out, but here lambda bar is a vector. So, you have to write it as c times lambda bar. And therefore, this implies that w bar the optimal vector w bar equals minus c minus half c lambda.

So this is basically the optimal vector w bar that is that expression for the 0. And again to find lambda bar use the constraint. What is our constraint? Well, our constraint is remember c bar c transpose w bar equals e bar 1. So, our constraint is c transpose w bar equals e bar 1, this implies c transpose no substitute for w bar minus half c lambda bar equals e bar 1 which implies minus half c transpose c lambda bar equals e bar 1.

(Refer Slide Time: 22:14)



Because we only need minus lambda bar over 2 this implies minus lambda bar over 2 equals c transpose c inverse e bar 1. That is the expression for your this is the expression for your, this is the expression for the luggage bottom lamp. In fact, it is an expression for lambda bar divided by. Now substitute this in the expression in if you call this one substitute this in one. You have w star now the optimal zero-forcing beam forming vector. You can write this as what is this?

This is basically you can take this factor of minus half inside. So, this will be I think it is just the c into r divided by 2 which is equal to minus lambda bar divided by 2 is this expression. So, this is simply c c transpose c inverse e bar 1. So, that is your optimal zero-forcing ok. So, this is the zero-forcing beam former which basically places a null in the direction of interfere. In the sense, I had completely blocks completely blocks the interference from the interfering user all right. Or it 0's the interference from the interfering users.

And in fact, this also on the one of the; it is also a very popular technique that is employed in practical wireless communication systems, especially in the presence of a large number of interfering users. As I already told you before, you can also use this in a cognitive radio scenario very have a secondary user there is an ongoing the ongoing primary user transmission. So, you can block at the second user receiver can block this interference caused by the primary transmitter by using zero-forcing wave forming.

So, because of it is low complexity also it tends to be one of the popular beam forming techniques, along with the interference the beam forming in the presence of interference that we have seen trees that is also termed as a cap on beam former alright. So, the zero-forcing beam former is also one of the popular modes of being forming that is employed in practical scenarios, alright. And this gives you a neat procedure to derive the expression for the zero-forcing beam, alright.

So, we will stop here and continue in the subsequent (Refer Time: 25:11).

Thank you very much.