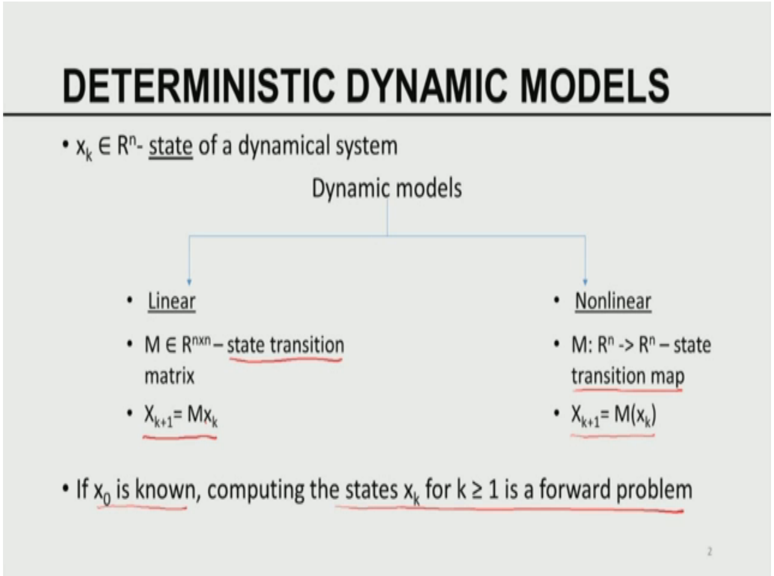
Dynamic Data Assimilation
Prof. S Lakshmivarahan
School of Computer Science
Indian Institute of Technology, Madras

Lecture - 20
Inverse problems in deterministic

We are now going to be talking about inverse problems for deterministic dynamic models. In particular we are going to be talking about the well known 4 dimensional variational assimilation method called 4 DVAR, which is also otherwise known as first order adjoint methods. Before starting on statement of deterministic dynamic model based inverse problems let us quickly review where we are and how we got here we first started with what is data assimilation why data assimilation. Then we looked at the mathematical preliminaries from final dimensional vector space matrix theory, then a multivariate calculus then optimization theory.

After that we introduced to the broad class of static inverse static deterministic inverse problems then we talked about matrix methods as well as direct iterative optimization minimization methods for solving static deterministic inverse problems. That is the sequence of topics we covered that nee natural leads to an extension of deterministic models, but instead of static models we are going to be talking about dynamic models, we are going to be talking about inverse problems that occur within the framework of deterministic dynamic models.

(Refer Slide Time: 02:02)

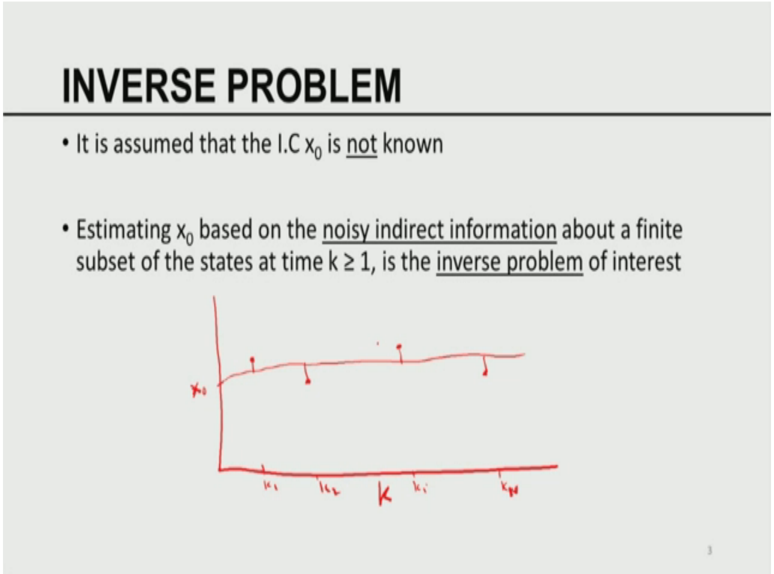


So, it is useful to talk about that useful classification of dynamic models let X k be the state of a dynamical system dynamical system is something that evolves in time as supposed to static models that is that remains constant with respect to time. Dynamic models can be either linear or non-linear in a linear model, the state X k evolves according to X k plus 1 is equal to M times X k where M is a n by M matrix. It is called the state transition matrix. So, the dynamic models are essentially given by the transition matrix M and an initial condition x naught if the matrix M is given and x naught is known computing X k plus 1 using M times X k is the forward problem, but here we are going to consider the inverse problem that assumes x naught is not known.

So, that is where the inverse problem comes in. So, I would like to be able to distinguish between inverse problem and a forward problem in the dynamical context. If x naught is known computing the states is called the forward problem for the linear model likewise for the non-linear model I should be given yet a state transition map M is a vector valued function of a vector M is from R n to R n here the state at time k plus 1 is given by M of X k as supposed to M times X k that is the difference between linear and the non-linear models.

Again given X naught whether it is a linear model or a non-linear model computing the future states is a forward problem now we are interested in the inverse version within the context of dynamical models.

(Refer Slide Time: 04:17)



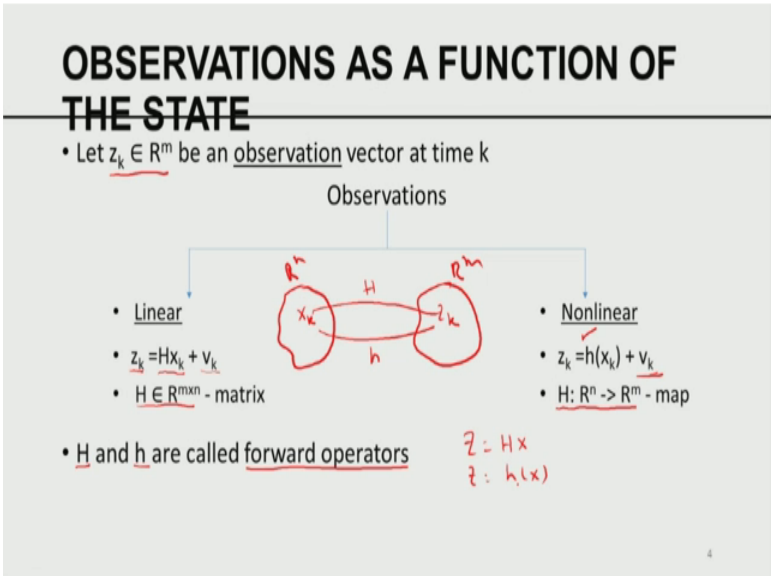
So, the in the inverse problem of interest to us it is assumed that x naught the initial condition is not known our goal is to be able to estimate x naught based on noisy indirect information about a finite subs sets of states at time k greater than one that is inverse problem of interest, I would like to pictorially represent this like this. This is k, I have been given observations at various instances in time k 1, k 2, k i k let us say capital noise; kn I have been given observations which are not directly state, but some functions of the state. So, I have been given some observations at this time at this time at this time at this time our aim is to be able to find the initial condition x naught.

Such that the sequence of state computed using a dynamical model is such that the model trajectory fits the observation in the least square sense and what is meant by least square sense, if I fixed x naught the model solution can be computed using the given model which is either linear or non-linear. So, that is a forward problem. So, given x naught compute the forward solution given the forward solution and observations which are induct measurements of this state, we get the difference between the observation and the state the observation and the state observation and the state observation and the state.

We are going to compute the sum of squares of the errors between observation on the state you can readily see the sum of squares of errors is the function of x naught because once x naught is given the solution is given for all time therefore, our job is to be able to adjust x naught such that the solution calculated from x naught best fits the observation in the sense of the sum of the squared errors is a minimum. So, that is the inverse problem of interest; that means, given information about the solution at time k greater than 1, I would like to be able to infer the state at time 0 that best starting from which the solution best matches the observations given.

So, the basic idea is essentially the same we are still going to pursue the least square framework. So, we are going to in we are going to formulate the least square criterion and look at ways by which I we can minimize the least square criterion the techniques are slightly involved that is largely due to the fact that the model is not static by dynamic. So, the basic principles of dynamic inverse problem is not much different from that of the static inverse problem except that we need to be able to handle the nuances exhibited by the dynamics.

(Refer Slide Time: 08:10)



So, we saw model can be linear or non-linear now we also said we have been given a finite subset of observations the observations themselves have to depend on the state if the observation do not depend on the state, then it does not have information about the state that is what we said that observation contain indirect information about the state. It is from this indirect information our job is to be able to extract information about the initial condition that is the inverse problem of interest in here again there are 2 cases to consider the observation vector zee with now functions of time. So, in the static problem we said zee belong to R m. Now we say zee k belong to R m what is zee k zee k is the observation at time k.

So, observation at time k may be a linear function of the state matrix H is i M by n matrix, but observations are now assumed to be corrected by noise V k is called the observation noise. So, this is the case of linear observation or the observation or linear functions of the state in in the non-linear case, I have been given a non-linear map H. H is a map from R n to R m. So, zee k is equal to H of X k we are going to again consider additive noise V k; V k or the observation noise it is generally model as mean 0 Gaussian noise with the pre specified covariance structure in the context of geophysical applications the matrix H and the map H are called forward operators.

So, you can think of the model space which is R n you can think of the observation space which is R m the model state is X k the observation is zee k, I can relate this using H or I can relate this using H 1 is the linear map another is the non-linear map. So, the information about the state is not directly known when H is an identity matrix then the observations are really measurements of the states if H is not an identity matrix or the map H is not a simple function the observations are complicated functions of the state of the system it is from this complicated information we have to unwind the information about the state using which I can estimate the initial condition by formulating a suitable inverse problems you can readily see in the case of a static problem.

We considered zee is equal to Hx as a linear problem we considered zee is equal to H of x as a non-linear problem the only difference between the static problem and the dynamic problem is that we have to keep track of that time in disease at which observations were given to us. So, that is the primary difference between the static and the dynamic cases.

(Refer Slide Time: 11:52)

EXAMPLES OF h(·) FUNCTION

State x	Observation Z	h(·)
Sea surface temperature, T	Radiance E measured by a satellite	Planck-Stefan's law $E = \alpha T^4$
Rate of rain fall	Reflectivity from the cloud measured by radar	Empirical nonlinear law
Speed of car	Voltage in cruise control	Faraday's law (Linear)

Now, I am going to give examples of function H just to give you a feel that this formulation is for real and it captures the spirit of what happens in practice, suppose I am interested in modeling the sea surface temperature of an ocean. So, let T be a sea surface temperature for example, in the analysis of el nino, we are interested and understanding the changes in the sea surface temperature at the equatorial pacific west of a Hawaii it is very difficult to go and make measurements.

So, we tried to infer the temperature of the sea surface by looking at the energy radiated from the hard surface by measuring the radiated energy from a satellite and I am recovering the temperature. So, if temperature is a state variable temperature T is a state variable if the observation zee is measured the observation is the energy E measured by the satellite the observation zee and the state are related by the equation called a Planck Stefan's law which is given by E is equal to alpha times T to the power of 4.

So, the status x the state x is T the observation zee is E and the function H the forward operator H takes the form E is equal to alpha to the power alpha times T to the power of 4, in another problem your yours models state variable could be the rate of rainfall, I would like to be able to predict how much rain will occur the next 24 hours. In order to estimate that I need to understand and make a measurement of the rate of rainfall often times their rate of rainfall the state variable is not directly observable as in the case of the energy radiated.

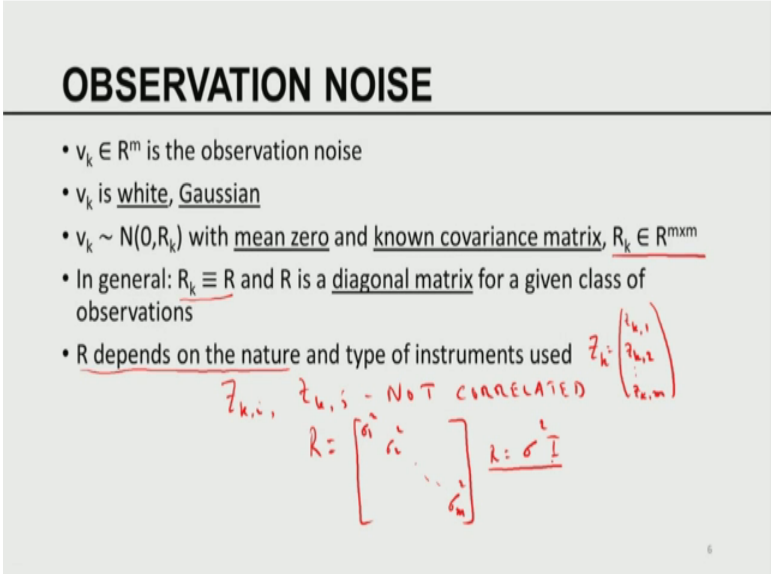
I can only measure indirectly the indirect measurements comes on reflectivity from the cloud as measured by the radar the radar sends out the beam towards the cloud the cloud consists of lot of droplets the droplets reflect the energy the energy radiated the radiated the reflected energy received by the radar is the measure of the amount of water content in the cloud the relation that relates the reflectivity to the rainfall rate is an empirical is an empirically derived non-linear law. This law was essentially derived at the national severe storms lab who specialize in applications of radar meeting to meteorological problem essentially a radar meteorology this was this was obtained as a result of an empirical curve fitting.

Another example could be if I in the design of cruise control of cars I would like to be able to measure the speed of the car speed of the car essentially measured in terms of voltage generated they put a voltmeter on a shaft that rotates. So, the voltage generated is proportional to the speed the depth the relation between voltage generated and the speed of a car is given by faradays law. So, you can see how the real states of the model and the actual observations are related using these examples I have given examples of the forward operator in the first 2 cases, it is non-linear the third case is linear.

So, identifying the forward operator H is not a trivial problem it depends on what is the state variable of interest to you in your analysis how the state variable can be measured is it directly measurable or it can only be in indirectly measured for example, pressure temperature wind velocity these are all directly measurable state variable, but if your model consists of state variables such as vorticity; vorticity cannot be measured directly. So, vorticity has to be inferred from certain observables that inference is done through certain well known mathematical relation those mathematical relation constitute the forward operator in question.

So, I would like to now summarize I have a model; model could be linear or non-linear I have in indirect information about the state; from the observation the observation could be again a linear function of the state or a non-linear function of the state.

(Refer Slide Time: 16:58)



We talked about observation noise V k when we dealt with the static problem even though we recognize that you can never make an observation without an error while we were cognizant of the fact that observations have errors we purposefully to simplify the problem assume that there is no error we simply close our eyes to reality and consider it to be a deterministic case.

So, in principle observations are riddled with errors one way to be able to model the error right from the days of gauss is to be able to model the observation error as a white Gaussian noise; that means, at a given time, the noise is Gaussian distributed, but if you consider 2 different times there is no correlation there is no temporal or spatial correlation between the observation noise that affect the observation at 2 different times are 2 different points in space these are the key assumption that are made to be able to simplify your analysis . So, V k is the observation noise that affects the observation at time k. So, V k is a vector of random variables it has 0 mean and a covariance Rk, it is assumed the covariance matrix is known Rk is a covariance matrix of size M by M it is also assumed Rk is symmetric and positive definite in most of the cases.

In general we will assume Rk is R; that means, our instruments are identical. So, even though I may be making measurements at various times the instrument that make the measurements in different times are identical. So, their error properties are same. So, I am going to assume Rk is R with a loss of generality one further simplification is that Rk is a diagonal matrix for a given class of observations. So, what does this mean I have zee k as my observation? Zee k is a vector which is zee k 1 zee k 2 zee k m. So, what is that we are going to assume zee k 1 has its variance zee k 2 has its variance zee k M has its variance but zee k 1 and zee k 2 are uncorrelated zee k i; that means, the ith component of the observation and the jth component of the observation are uncorrelated not correlated.

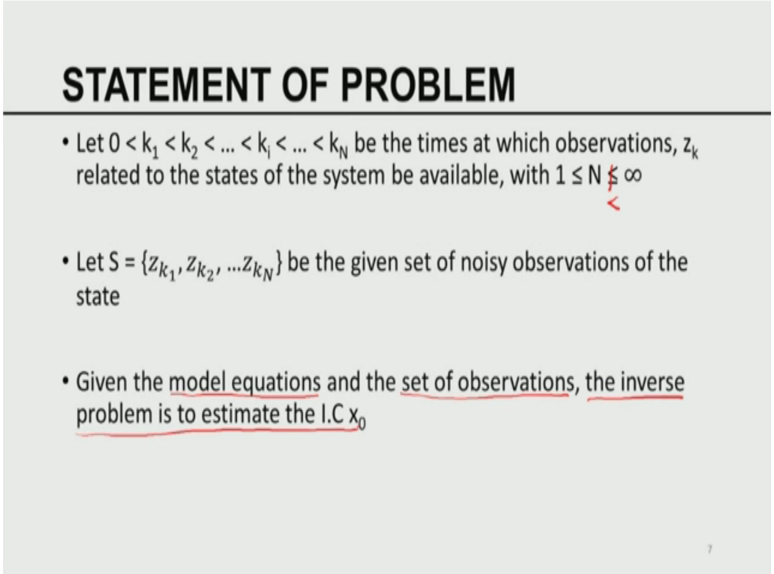
This implies that the matrix R is essentially a diagonal matrix sigma one square sigma 2 square sigma M square where sigma i square is the variance associated with the measurement of the ith component of the observation, but these are simplifying assumptions these assumptions are pretty close to reality again I would like to emphasize the variance depends on the nature and the type of the instruments used in the observations. So, here comes the challenge when it comes to satellite we generally do not know too much about the error properties especially if a satellite has been around for a long time the proper the measurement the quality of the measurements change in time.

So, one probably has to make approximations one approximation that is generally made is that Rs is equal to sigma square i; that means, it is a diagonal matrix where all the diagonal elements are the same; I am not going to distinguish it because I just do not know if you do not know something, but still you want to be able to include the nice covariance one way to assume is a simple case is that R is a diagonal matrix with a constant diagonal elements. So, what is it what is the implication all the measurements are equally accurate no 2 component of the measurements are correlated and. So, this is a much simplifying assumption.

So, one can assume R to be a general symmetric positive read matrix to the level of being R is equal to sigma square i. So, that is the range of variation these range of variations are allowed because we generally may not know the error properties of observation from radar we may not know error properties of observation from satellites we may not know error properties of certain buoys if they have been around for long after a long time.

So, that is the dilemma that one has to deal with. So, we want to be as close to reality without making the problem too difficult to solve. So, the compromise here are covered in these basic assumptions about the observation noise.

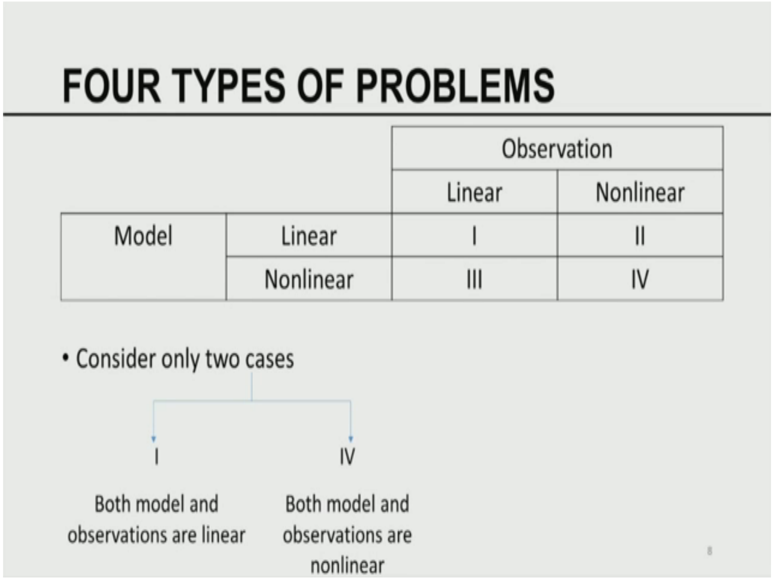
(Refer Slide Time: 22:16)



So, now, I am going to formally state the problem let k 1, k 2, ki kN be the n times at which observations are available for n being some finite number it could be 1 2, I do not think it should be less than or equal it to be strictly less than infinity finite now finite number of observations. So, given a finite set of n observe noisy observations of the state. So, these are noisy observations about the state means what this now this may be direct measurements it may be indirect measurements we just do not know, but we should be ready for it. So, zee contains information about the state either through the matrix H or through the map H the non-linear map H.

So, given the model equation. So, here is the problem given the model equations linear non-linear and the set of observation the observation could be linear or non-linear the inverse problem is to estimate the initial condition that best fits the observation that is the statement of the inverse problem in here.

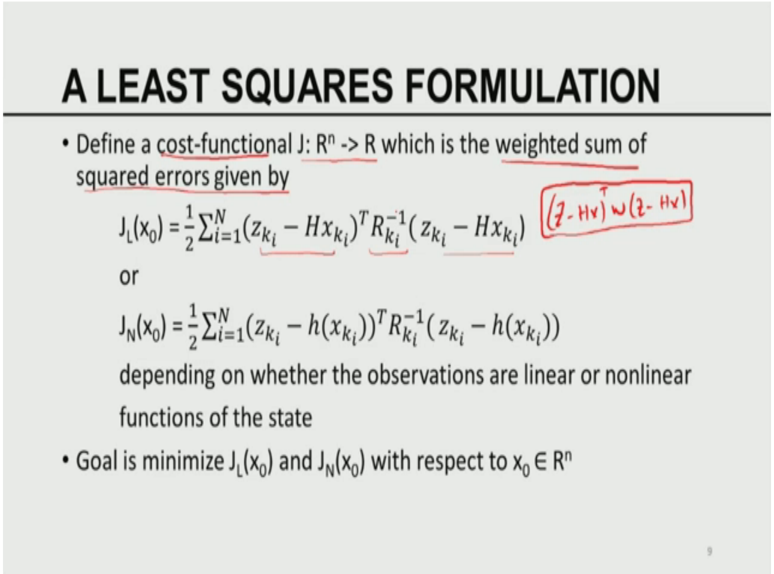
(Refer Slide Time: 23:33)



So, before I go further there are 4 different formulations of this inverse problem the model could be linear non-linear observation could be linear non-linear in case one both model and observations are linear in case 4 both are non-linear the other 2 one is linear another is non-linear. So, the simplest possible case in both are linear one the most complex case is 4 which is both the model and observations of non-linear. So, we consider these 2 extreme cases if we understand how to solve these 2 problems the extreme cases the intermediate case 2 and 3 would become obvious I would like to I would like to open up the class of all problems that one could consider. So, that is what we have we have done this far, we have essentially stated all the basic information one needs.

So, please remember we need ton of information I need to know the model, I need to know the function that relates observation to the state we need to know properties of the observation noise, we need to know the times and the values of the observation at specific finite number of times given all these we can then state the inverse problem in a most in a more formal way as we will see right now.

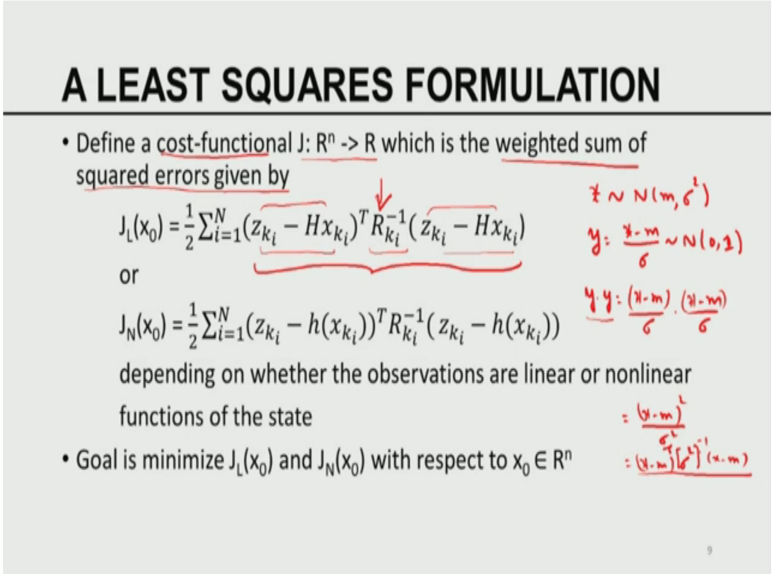
(Refer Slide Time: 25:05)



So, I am going to start with case one model is linear observations are also linear functions of time, I am sorry linear functions of state not time state, I am sorry.

So, we are going to define a cost functional the cost functional is J the J is a scalar valued function of a vector J is the weighted sum of squared errors J is given by zee ki minus H X k i transpose R ki inverse zee ki minus x ki this looks like this should be very familiar to all of us we considered zee minus H of x transpose w zee minus H of x this is the weighted version of the least square criterion that we use for the static problem instead of zee I have zee k i instead of x i have x ki instead of w i have Rk i inverse I am simply still considering R is depend on time why is this kind of a waiting I would like to talk about it for a moment before I go further to that end let me consider what we normally do in basic statistics.

(Refer Slide Time: 26:45)



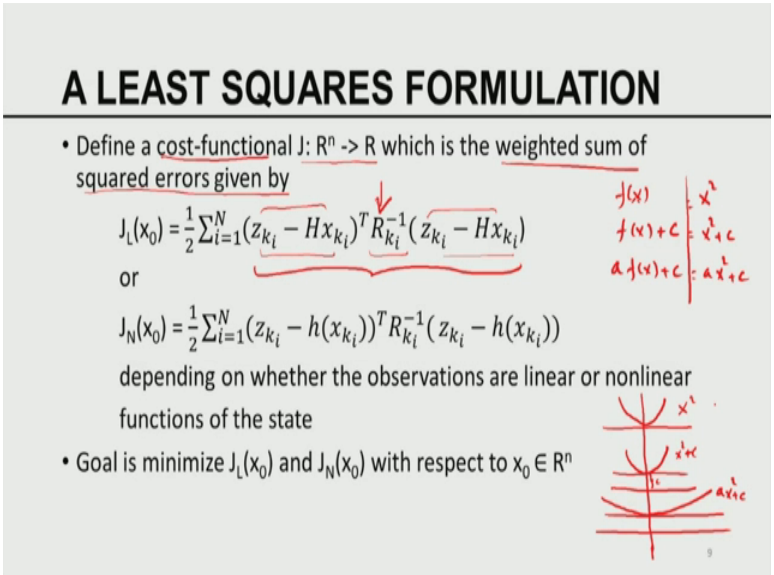
Let x be a normal random vector with mean M and variance sigma square if I want to normalize this I am going to consider a new random variable y which is equal to x minus M divided by sigma because I am divide; I am subtracting M y is going to be normal with 0 mean and unit variance. So, subtracting the mean and dividing by the square root of variance is essentially normalization therefore, if I want to be able to consider y times y this will be x minus M divided by sigma times x minus M divided by sigma which is equal to x minus M square divided by sigma square which I can write as x minus M times sigma square to the power minus 1 x minus M x minus M is a scalar sigma square is the inverse of the variance x minus M is a scalar.

So, without loss of generality the transpose of a scalar is a scalar; I can put a transpose here. So, you can readily see x minus M transpose sigma square inverse x minus M what is that it is a square of the normalized random variable with 0 mean and unit variance.

So, this is the form that is replicated here. So, divide. So, this is the error this is the error this is the error this is the weight function the weight function in here is. So, chosen that it normalizes to be similar to what we have done. So, we are interested in a normalized form of the error in the error which is a measure of the difference between zee and hx zee and hx is call the residual in our old notation from static problem. So, what is that one you can you can see this is the weighted sum of squared errors or square residuals; I am using the word error and residual in the same fashion and that is what it is I have this quantity at each time instant I have n time instance at which the observations are available. So, I am summing this up from 1 to n, I have added a multiplying factor one half and that is a technical thing I will explain that in a moment before we go further I hope the reason behind the choice of the weight function which is the inverse of the covariance matrix is clear from this basic idea of normalization of errors.

Now I would like to I would like to be able to talk about why half to understand half I want to be able to go back to in some basic principles let me make some space by erasing some of the unwanted things from here. So, let us suppose I have a function f of x.

(Refer Slide Time: 30:27)



Let us suppose; I have the function f of x plus c as it a; suppose I have the function 8 times f of x plus c. So, f of x could be as an example could be x square. So, this is equal to x square plus c this is equal to eight times x square plus c; let us graph these functions now.

This is x square, if c is positive this is c this is x square plus c if c is positive and a is positive this is going to be this is going to be a x square plus c. So, let us assume a is 2 now look at this. Now I have 3 functions which are one is x square another is x square plus c another a times x square plus c even though the shapes of the functions are different you see the point at which the minimum occurs has not changed; that means, the minimizer the minimum value of x or the x that minimizes f of x for f is also the same minimizer for f of x plus f of x plus c is also the same minimizer for eight times f of x plus a. So, what does it mean if I multiply a function by a constant the location at which the minimum occurs does not change if I add a constant to a function the location at which minimum occurs does not change if I add a const; if I multiply by a constant and add a constant the location at which minimum occurs does not change; that means, the minimizing value of x is invariant with respect to addition of a constant multiplication by a constant. So, given this property I am multiplying this function the summation by half. So, the half plays the role of a why do I add a half in here the add a factor half in here.

You can see zee minus H of x transpose Rk inverse zee minus H of x that is a quadratic function when you differentiate the quadratic function to compute the gradient there is a annoying factor 2 coming in the 2 can be cancelled. So, it is simply an algebraic simplification in the ultimate formula that I would get; anticipating that I do not want to deal with that factor 2, when I differentiate quadratic functions I simply added the 2 to simplify some of the expressions.

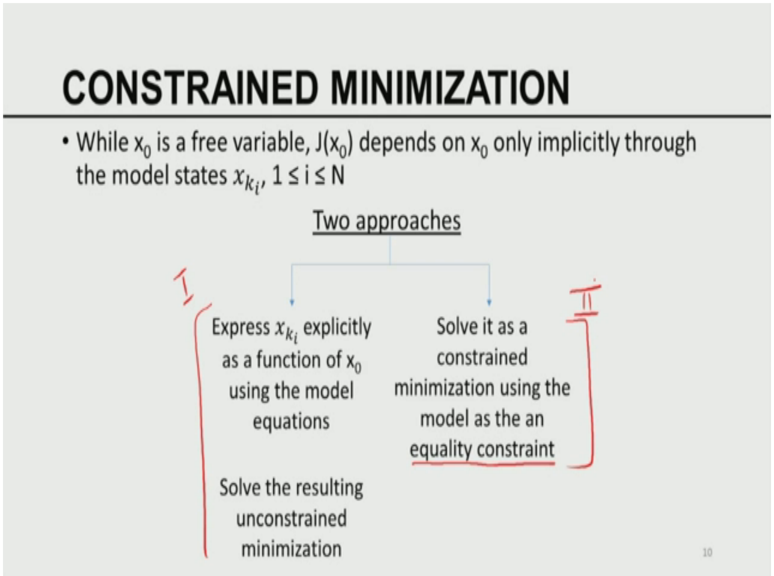
So, multiplying by a constant does not change the result adding a constant does not change the result, without loss of generality I can multiply a function to be minimized by any positive constant without altering any of the properties of the minimizer, we would like to. So, J L is the weighted sum of squares in the linear case J N is the counterpart for the non-linear case. Now look at the difference now the difference is essentially in this term, the difference is essentially in this particular term, you can readily see the this term converts to H of x this term converts to H of x everything else remains the same.

So, if the states are non-linear if the observations are non-linear function of the state I use J N if the observations are linear functions of the state I use J L. So, we have an expression to minimize. So, this is. So, the value of J is a measure of the set when J takes the minimum the fit is the best. Now J is a function of x naught on the right hand side, you do not see x naught directly on the right hand side x naught comes indirectly through X k; X k is the state of the system at time k, but X k is a state of the dynamical system the state of a dynamical system at any given instant is a function of the initial state. So, X k depends on x naught implicitly.

So, I would like to add a comment to say that J L of x naught or J N of x naught are implicit functions of time and our job is to be able to minimize this implicit functions of time I am sorry implicit functions of the state the state being the initial state that is the that is the challenge associated with this formulation in the dynamic case.

Now, let us go back. So, I am going to have to minimize this with respect to x naught x naught does not occur, but x not occur implicitly through X k; X k is related to x naught through the model therefore, the model equations are inherent within this expression; that means, that means X k are not free variables X k are related to x naught through the model. So, the model is to become is to become constraint is to be considered as a constraint and that is what our next.

(Refer Slide Time: 36:30)

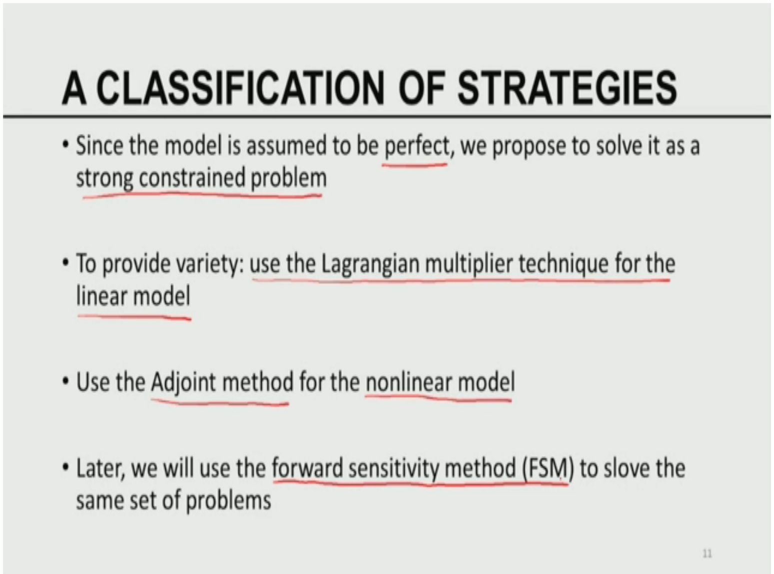


Realization is all about while x naught is free J of x naught depends on x naught only implicitly through the model state xki that what we just talked about. So, there are 2 approaches to handle this implicit minimization problem one is to express xki explicitly as a function of x naught using the model equation and solve the resulting unconstrained minimization problem that is approach one that is the approach one that is the approach one or else you would like to be able to solve this as a constraint problem where we are going to force X k to be related to x naught through the model equations. So, the model equations would constitute the so called equality constraint.

these are the 2 equivalent ways of looking at this minimization problem these are 2 approaches one can utilize both the approaches are meaningful; however, this approach on the left the first approach, I would like to call is more suitable for linear system this approach second it is suitable for both the linear and non-linear systems, I hope the formulation is very clear. Now I have to minimize J L of x naught or J N of x naught as an implicit minimization problem where you relate X k to x naught through the model equations.

So, either you use the model express X k in term of x naught explicitly and solve an explicit unconstrained problem or simply use the model as a constraint and formulate it as an equality constraint problem we already know equality constraint problem are solved using Lagrangian multiplier we have already come across the use of Lagrangian multiplier constrained optimization problem in the context of under determine static deterministic inverse problems. So, the techniques are very familiar to us. So, the same kind of techniques are going to be applied in the dynamic case as well. So, you can see the methodology is very similar between static and dynamic problems except that in dynamic problem we simply need to be able to take time into account.

(Refer Slide Time: 39:14)



So, a comment about some classification of strategies for solving this problem whenever we assume the model is deterministic; what is the implicit assumption we assume the model is perfect.

So, in any analytic problem when we assume the model to be deterministic there is an implicit assumption about our belief of the model being the best, but I know is the best or I have utilized the best model known to mankind there is nothing better than available.

So, I am going to assume the model is perfect; if the model is perfect if I want to be able to use the model as a constraint I would like to force the constraints strongly. So, I would force the problem as a strong constraint strong equality constraint problem you may remember strong constraint and weak constraint we have already talked about within the contrast of constrained optimization.

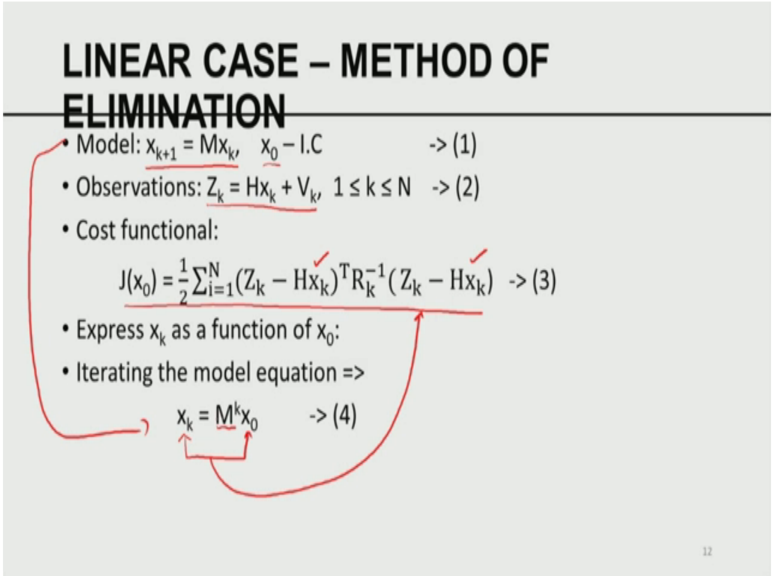
So, perfect model assumptions. So, let me underlined this now perfect model assumptions and strong constraint formulation why strong constraint formulation if I assume the model is perfect I want the model to have the final say in what I do because I believe a model. So, much that model has the last say to provide a variety we will use Lagrangian multiplier technique to solve the linear model problem as a strong constraint problem. So, you can see there are number of formulations.

The formulations of the problem the model assumption linearity nonlinearity, we can see the tree of potential ways to be able to mix and match. So, I will use Lagrangian multiplier techniques as a strong constraint formulation to solve the linear model for the non-linear model I will use the so called adjoint method why do we use different techniques to solve different versions of the problem to be able to expose the richness of the class of methods that has come to be known for solving these kinds of dynamic problem.

Later we will also say solve the same problem using a new class of method called forward sensitivity method which was developed by as some five six years ago. So, you can see the variety of techniques now model being perfect strong constraint using Lagrangian multiplier model being perfect non-linear case using adjoint method model being perfect we will use forward sensitive method to solve the same set of problems.

So, I am going to illustrate 3 types of techniques to solve the inverse problem within the context of the assumption that the model is perfect I am going to use it as a strong constraint.

(Refer Slide Time: 42:22)

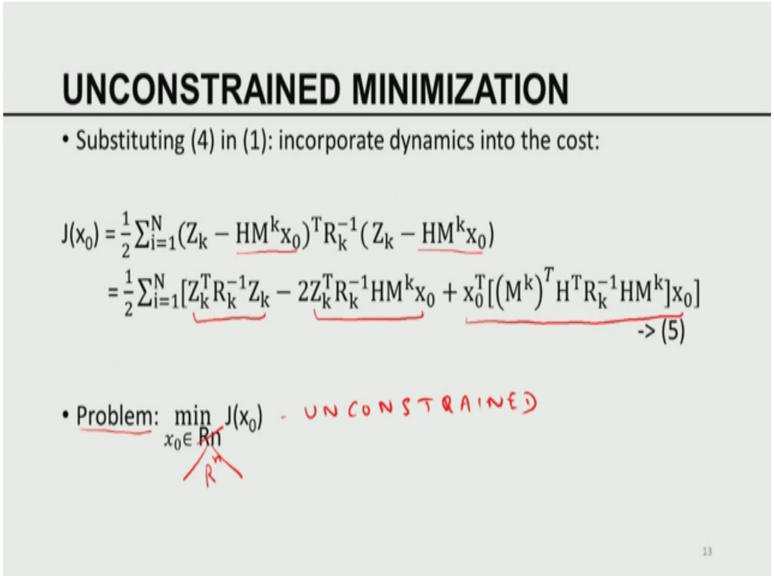


So, let us start with the linear case to get the ball rolling in the solution process. So, model is X k plus 1 is equal to M of X k x naught is a initial condition observations are zee k is equal to H of X k plus V k the cost functional is given by this. So, what is the easiest? So, before I utilize Lagrangian multiplier techniques and other things I am going to now simply use the model to be able to express X k in terms of x naught the simplest possible exercise this is simply method of elimination you can you can you can think of it.

So, I express as a function of x naught by iterating the model you can readily see X k is equal to M to the power k x naught. So, I have expressed here solution model state at time k to the model state at time 0 through the kth power of this state transition matrix; yes multiple M to the power of k involves matrix; matrix multiplication we know matrix matrix multiplication is very expensive computationally even though computationally this could be very expensive; I want to provide a basis for comparison for later methods that is why we are going to indulge in this so called method of elimination.

So, look at this; now this is x k. So, x ks are needed here look at this. Now xks are xks are needed here sorry, x ks are needed here X k is given in here. So, I am going to use this X k in 3. So, substituting X k is equal to M to the power of k x naught in expression for 3.

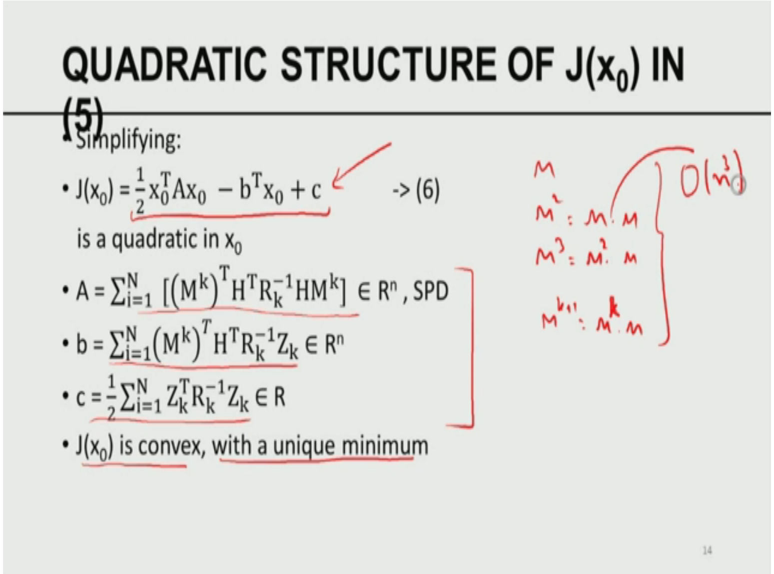
(Refer Slide Time: 44:30)



I get the following. So, this is H of X k with M to the power of k x naught H of M to the power of k x naught. So, the right hand side is an explicit function of x naught I can now multiply this is a constant term this is a linear term x naught this is the quadratic term in x naught even though the expressions are little bit more complicated than the ones in the static case essentially we are dealing with a constant term a linear term and a quadratic term. So, the essential structure has not changed even though the form of the expressions have changed my job is to be able to minimize x naught belonging to that it must be R to the power of n. So, minimize J of x naught where x naught belongs to R to the n.

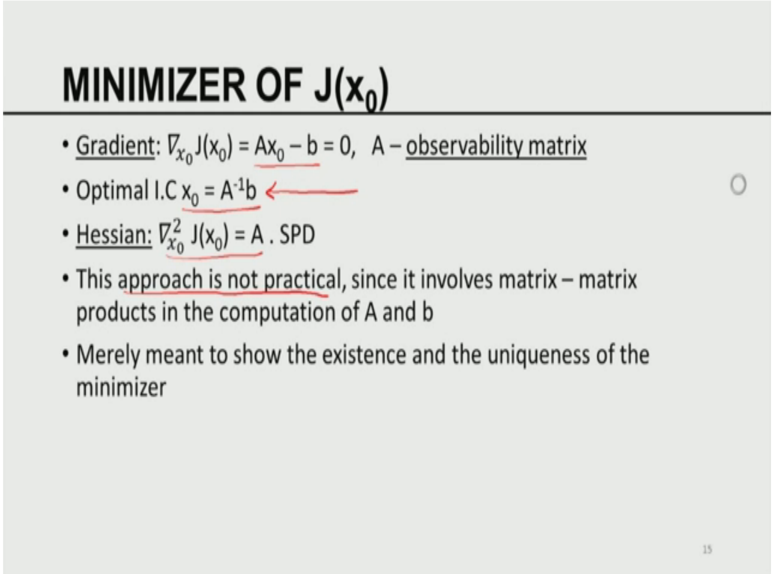
So, this is an unconstrained minimization problem unconstrained minimization problem. So, how do I solve this I compute the gradient of J with respect to x naught we know how to compute the gradient of a linear function we know how to compute the gradient of a quadratic function this is what we saw in the module on multivariate calculus. So, we are going to draw our exp from our experience in multivariate calculus and so, to be able to do that I am going to simplify this expression. So, that it looks much simpler than this.

(Refer Slide Time: 46:07)



So, I am going to rewrite that as a expression like this which is much more simpler in this case you can readily see a is given by this sum of these matrices b is given by some of these and c is given by this. So, I would like you to be able to look at this now J. So, this is a sum of quadratic function this is some of linear functions this is some of constant functions. So, I can express J x naught by using these substitutions defining a b and c express J x naught as follows I am sorry J x naught as follows. So, this is the expression where the new variables a b c are given in here if I the reason for writing at a six because is the quadric function if a is symmetric positive definite J naught is convex if J naught is convex again from basic optimization theory we know the unique minimum exists.

(Refer Slide Time: 47:12)



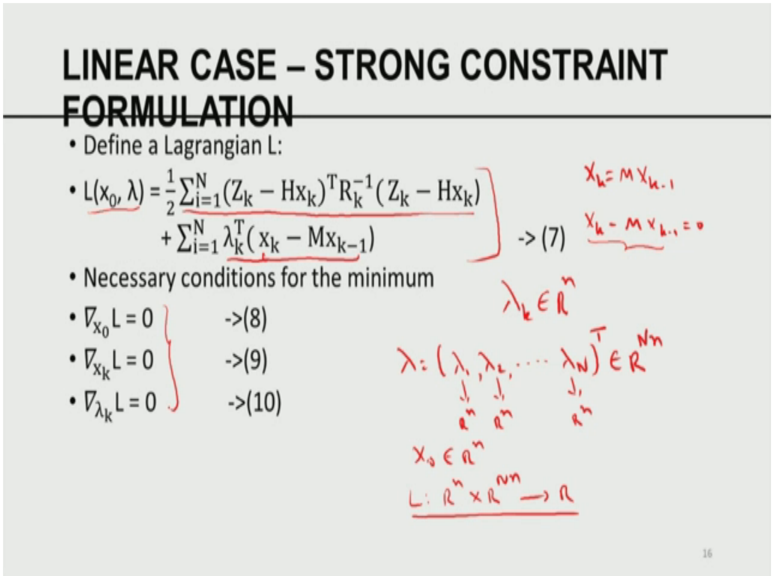
I can compute the gradient which is equal to A x naught plus A x naught minus b and that is equal to 0. So, I get the optimal x naught to be a inverse b the hessian is essentially a which is SPD. So, this, I have essentially I have essentially solved the problem and I have computed the solution of form of the solution the optimal initial condition is given by this even though I have an expression for the optimal initial condition this approach is far from being practical why because it involves matrix; matrix product. Let us look back that a matrix it consists of M; M to the power k transpose and M to the power k.

So, if I have a matrix M if I have a matrix M; M square M cube M to the power k plus 1 is equal to M to the power k times M. So, we are going to have to compute these things recursively each one involves matrix; matrix multiplication this is M square times M and so on. And matrix matrix multiplication takes of the order of n cube and yesterday we saw n cube complexity when n has the order of the million would take a very long time therefore, this method while is giving you a very beautiful like mathematical expression it is far from being our practical.

The reason for doing this is not because it is practical, but because it brings the essence of the problem the essence of the problem is that when the model is a linear the observations are linear functions of the time the J function. So, concocted reduces to a convex functional a quadratic convex functional standard quadratic convex functional which is a unique minimum therefore, optimize are exists I can in principle give an expression for the optimizer. So, I have theoretically solve the problem, but this way is not practical I would like to now ask you to concentrate on the expression for the matrix a expressions of the vectors b as well as expressions for the constant z.

So, these are all computed using very many using all the properties I know M is the model H is the forward operator Rk is the nice covariance. So, you can see the model component the observation component the observational error component all these comes in a hue and it is this hue that makes the matrix a is a combination of various factors.

(Refer Slide Time: 50:05)



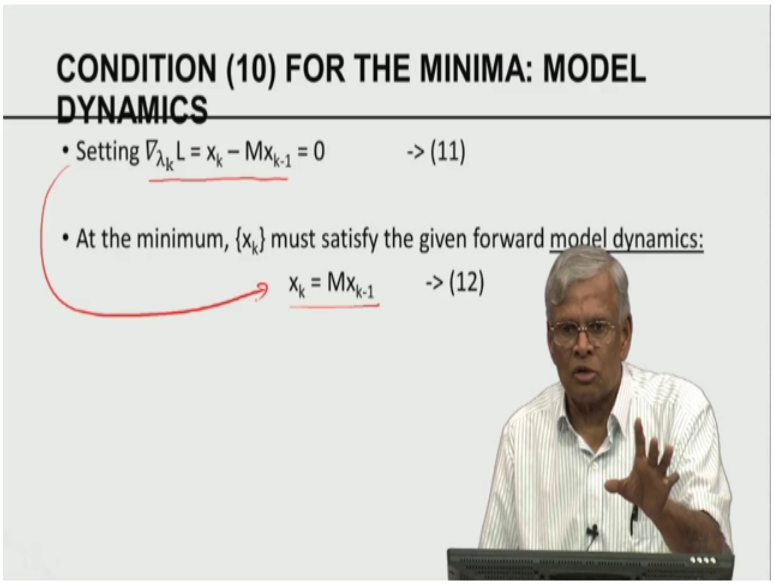
Having seen a simple method of elimination now I am going to come back to formulating, it as a Lagrangian multiplier based technique as a strong constraint Lagrangian multiplier based technique. So, to that end I am now going to construct a Lagrangian function. So, this is the objective function X k is equal to M of X k minus 1 that is the model. So, X k is equal to X k is equal to M of X k minus 1 is the model I am going to rewrite this as X k minus M of X k minus 1 is equal to 0 that is the constraint.

Now, please remember X k is the vector M is the matrix X k minus 1 is a vector. So, the whole thing is the vector this vector belongs to Rn. So, I am going to take the inner product of this vector by lambda k lambda k transpose X k minus M of X k minus 1 that is the Lagrangian multiplier term there is one such term for each k summation from one to n. So, lambdas are called the undetermined Lagrangian multipliers lambda k they are all vectors in R n. So, inner product that is constant these quadratics form that is a constant. So, L of x naught lambda that is a scalar function it is a scalar function of x naught and lambda when I say lambda lambda is not one lambda lambda is a collection of all the lambdas lambda one lambda 2 all they up to lambda n each of the lambdas are vectors in R n each of the vectors are R n therefore, I have n capital n vectors each of size little n. So, this lambda is this lambda is a vector that belongs to R n n.

Because each of them N n; there are n of them. So, that is the vector of that belongs to n n. So, you can see x naught belongs to R of x naught belongs to R of n therefore, L is a function that maps R of n cross R of N n to R yeah that is the function Lagrangian multiplier it takes 2 input the first input is the initial condition the second input is a collection of n Lagrangian multipliers each of which has dimension little; I hope the structure of the Lagrangian function is clear it is simply an extension of what we have done previously. So, once the problem is formulated as a strong constraint problem using Lagrangian multiplier the constrained optimization problem now is solved as an unconstrained optimization problem by minimizing L with respect to both x naught and lambda.

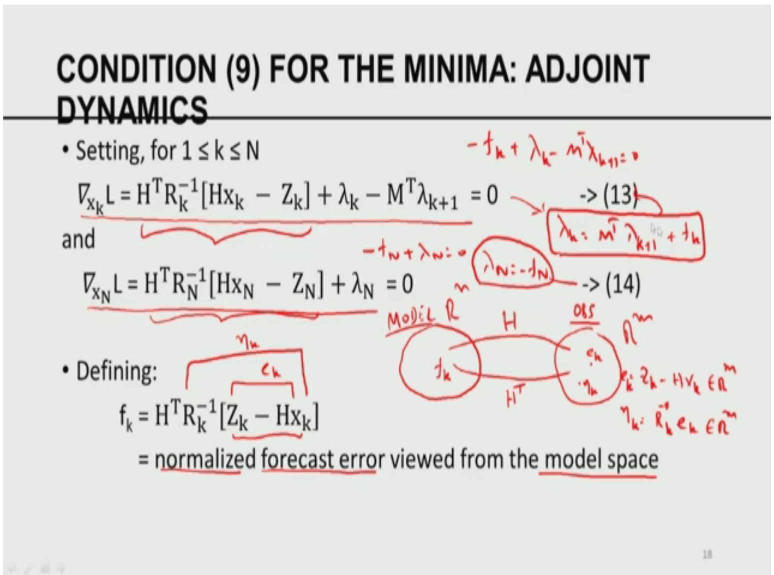
So, compute the derivative of L with respect to x naught compute the derivative of L with respect to X k compute the derivative which is the gradient of L with respect to lambda equate each of them to 0 this is a set of necessary conditions one has to satisfy for the minimum of the Lagrangian function.

(Refer Slide Time: 53:46)



By setting the gradient of lambda with respect to lambda k i get the model equation. So, X k minus M of X k minus 1 0 essentially means the model is satisfied. So, I am forcing the model as a strong constraint using 11 at the minimum X k must satisfy the forward model dynamics X k plus 1 is equal to M of X k minus 1. So, that is what is being given by this. So, this essentially leads to forcing the model as a strong constraint the model must always have the last say the model equation must always be satisfy in defining x xs are not free variables they are constrained by the model.

(Refer Slide Time: 54:31)



Now, I am going to compute the derivative these are this is the mechanics of Lagrangian method I have to compute the gradient of L with respect to X k and that gives rise to this equation it is very easy for me to write the equation 13, but it will take couple of minutes for you to be able to derive I am not going to spend time in trying to derive this equation from the Lagrangian, but it is an important exercise for you to be able to apply the basic principles of multivariate calculus to be able to compute the expression for the gradient of the Lagrangian.

Again a gradient for the Lagrangian in here equate all of them to 0 now we have to solve the system of 3 equations 12, 13 and 14. 12 implies, 11 is satisfied, 13 implies the gradient of L with respect to X k is 0 14 implies the gradient of L with respect to x xn is 0 in order to be able to solve this problem I want to be able to concoct a new variable I do not want to write too many things. So, I am going to combine several variables into one to that extent I am going to defining, yeah, yeah, yeah variable called f of k sorry I am going to defining a variable call f of k.

Now let us understand what f of fk is zee k minus H of X k that is the residual if zee k is equal to H of X k there is no residual the difference between the observation and the H of X k is the model counterpart of the observation actual observation minus the model counterpart of the observation is the residual or the error if I if I divide that by Rk minus 1 that is a form of a normalization I would like to now bring the operator H transpose we are looking at the first time.

So, this is R n this is rm. So, zee minus zee k minus H of X k belongs to R M it is in here I am going to call it ek then I am going to call it eta k which is equal to Rk inverse ek that is again belonging to rm. So, if this is ek this must be eta k. So, ek is the actual error eta k is the normalized error H is a map that goes from R n to R m H transpose is a map that goes from R m to R n therefore, H transpose eta k is f of k.

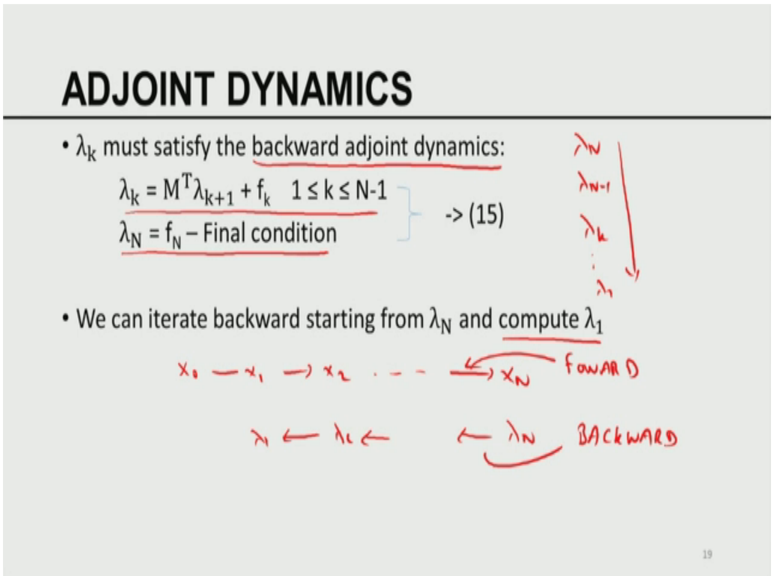
So, f of k. So, this is E of k; this is eta of k H transpose eta of k is f of k I hope that is clear therefore, we can interpret f of k as the normalized forecast error. So, zee k minus H of k is your forecast error dividing by Rk minus 1 is a form of normalization or waiting and multiplying that on the left by H transpose I am transferring from the model space to the ob from the observation space to the model space therefore, f of k is their normalized forecast error mu viewed from the model space.

So, it is very important to understand why we want to be able to convert the observation at the forecast error from one space and transfer to another space because all the computations are done in the model space.

So, this is the model space this is the observation space the data assimilation computations are done in the model space model space is R n therefore, when I have a quantity which is not in this space I have to bring the corresponding version of that quantity from observation space to the model space. So, that is what f of k is all about why is f of k is important look at this now let us go to 13 in 13 I have H of X k minus zee k. So, that is minus z k Rk inverse that is minus eta k H of transpose minus eta k this is minus f of k. So, you can readily see this quantity is minus f of k again this quantity is minus f of n.

So, that is the reason why we have defined this term. So, using f of k I can simplify the expressions for the derivative and handle it with little bit more easily I hope the notations are clear I hope the reason for defining these newer quantities is also clear. So, with this in mind I am now going to solve 13 and 14.

(Refer Slide Time: 59:36)



If I solve 13 and 14 let us go back; if I from 13 what do I what do I have minus I am I am sorry we let lets from 13 what do I have minus f of k plus lambda k I am sorry plus lambda k minus M to the power of M transpose lambda k plus 1 equal to 0 equal to 0.

So, how can I rewrite this this can be rewritten as lambda k is equal to M transpose lambda k plus 1 lambda k plus 1 plus f of k.

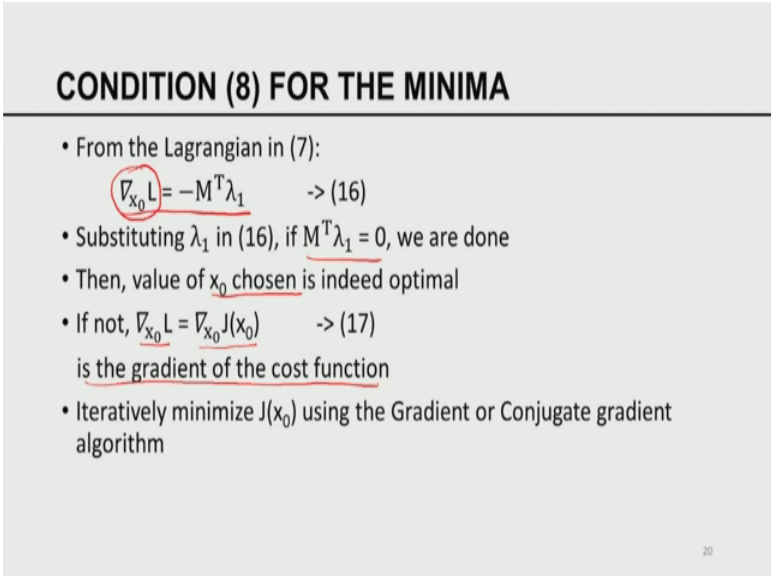
So, I am relating k plus 1 to k I am going backward in time I am going backward in time I just want to rewrite this and then explicitly write that k plus 1 this is k plus one. So, if 13 holds good this is the form 13 takes if 14 has to hold good the 14 takes the form minus f of n plus lambda n is equal to 0 therefore, lambda n is equal to minus f of n that is what 14 is all about.

So, this is 14 this is 13. So, 13 and 14 when solve give rise to these 2 equations. So, lambda n is equal to f of n please remember that that is what lambda n is equal to f of n that is 14. So, lambda k is equal to M transpose lambda k plus 1 plus f of k is 13. So, lambda k is equal to M transpose lambda k plus 1 f of k for one less than k less than n minus 1 lambda n is equal to f of n is the final condition therefore, we have to compute lambdas backward in time starting from lambda n which is which is the final condition this is called the backward dynamics.

This backward dynamics has come to be known as the adjoint dynamics that is the adjoint equation we can iterate. So, f fs are known M transpose are known fn is known. So, I can compute from fn to f of n minus 1 to from lambda n to lambda n minus 1 to lambda k to lambda one I can compute backward in time using this backward adjoint equation.

So, we can compute lambda one by going back now I would like to remind you 2 things the model is solved forward x naught to x one to x 2 all the way up to xn here is lambda n is lambda n minus 1 lambda 2 lambda one. So, these are considered backward the forward dynamics of the model the backward dynamics of that joint backward dynamics of joint this backward dynamic comes very naturally from the Lagrangian multiplier problem.

(Refer Slide Time: 63:10)



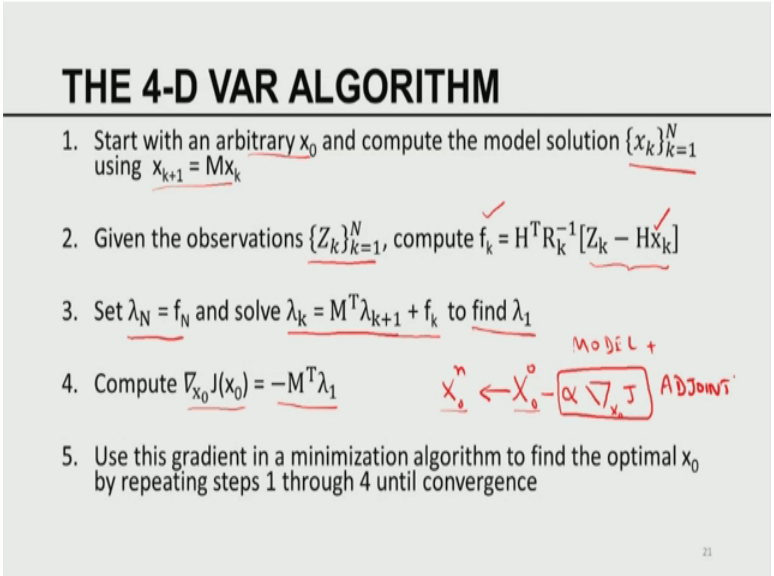
So, once lambda n is known froms from the Lagrangian seven we can readily compute the gradient of L with respect to x naught is minus M transpose lambda one. So, I computed lambda one by the backward adjoint equation I computed the gradient. Once I have computed the gradient we can do lot of things with that if this computed whereas, the gradient is 0, then I am at the minimum because at the minimum the gradient must vanish please understand I have converted the unconstrained minimization to a constrained minimization of L gradient of L with respect to x naught must vanish at the minimum gradient of L at the point x naught is given by minus M transpose lambda one lambda one was obtained by computing the backward dynamics.

So, so if it turns out let me go back if it turns out the x naught I chose from which I computed f x 1, x 2, x 3, x 4 from which I computed the errors from which I computed f ns from which I computed the lambdas from which I computed minus M transpose from which I computed minus M transpose lambda one and if this lambda one happens to be 0 well and good I am done.

But it stands reason to expect this calculation would not in general v naught equal to 0 because I chose x naught arbitrarily. So, for a arbitrarily chosen x naught in general need not be optimal if in general need not be optimal the gradient of L with respect to x naught under the constraint is the same as gradient of J with respect x naught I have computed the gradient of the cost function using the back adjoint method.

Once I have the gradient what is that I need to do; I can utilize either the gradient method or in the conjugate gradient method to be able to minimize it gradient method and conjugated methods we have already talked about in the previous modules. So, we are not going to repeat many of those things we already know. So, the emphasis in here is essentially computing the gradient to the cost function using forward run of a model and the backward run of the adjoint. So, what is that how is this algorithm work.

(Refer Slide Time: 65:55)



The algorithm works as follows this algorithm as come to be call the 4 DVAR algorithm this also call first order adjoint method by whatever name you call the essence of the method goes as follows.

Start with an arbitrary x naught. So, this is the algorithm 4 DVAR algorithm start with an arbitrary x naught you are given a model compute the sequence of state computed by the model you are given a set of observations. So, from the model you know x naught X k you already know H you already know zee k. So, you can compute z k minus H of X k that is a forecast error you compute the normalization you compute the model counterpart of that. So, f of k can be computed explicitly. So, run the model forward given the observation you can essentially compute the quantities of k set lambda n is equal to f of fn solve lambda k is equal to M transpose lambda k plus 1 plus f of k iterate backward in time compute lambda one compute minus M transpose lambda one that gives you the gradient use this gradient in a minimization algorithm to find the optimal x naught by repeating the steps one through 4 until convergence.

So, what does it mean you start with an x naught sorry you start with an x naught you start with an x naught you compute the gradient with respect to x naught of J you go in the negative of the gradient alpha times J of x naught then you say x naught is new. So, this is x naught old you compute the gradient you subtract alpha times gradient of J define the new x naught then start with this new x naught run the model forward in time calculate the forecast errors compute the backward and repeat this loop.

So, there are 2 loops I want you to be able to recognize one is the optimization loop another is the condition computing the gradient loop one loop feeds into the other. So, to be able to go from one initial condition value to the next initial condition value I need to run the model one squadron time and the adjoint backward in time. So, this requires the model run and the adjoint run.

Plus adjoint run, I am sorry, adjoint run model run and the adjoint run. So, you need to do both to get one gradient if you get one gradient you can get the new operating point and you repeat the cycle until convergence that is the key a that is the essence of the 4 DVAR algorithm you can readily see this algorithm is extensive in terms of using the model in the observation it provides you incremental improvement for their cost function if I use gradient algorithm and the objective function is convex we have already seen gradient function converges only asymptotically if the objective function is convex and quadratic if I use a conjugate gradient methods I can hope to get good convergence by finite number of operation finite number of a iterations.

So, we are going to be concerned with the general utilization of all the principles we have seen. So, far namely optimization algorithm on one hand model running on other hand adjoint method on the other hand all comes into a hue to make this algorithm called 4 DVAR algorithm that is the summary of 4 DVAR algorithm for the case of linear deterministic dynamic system.

Thank you.