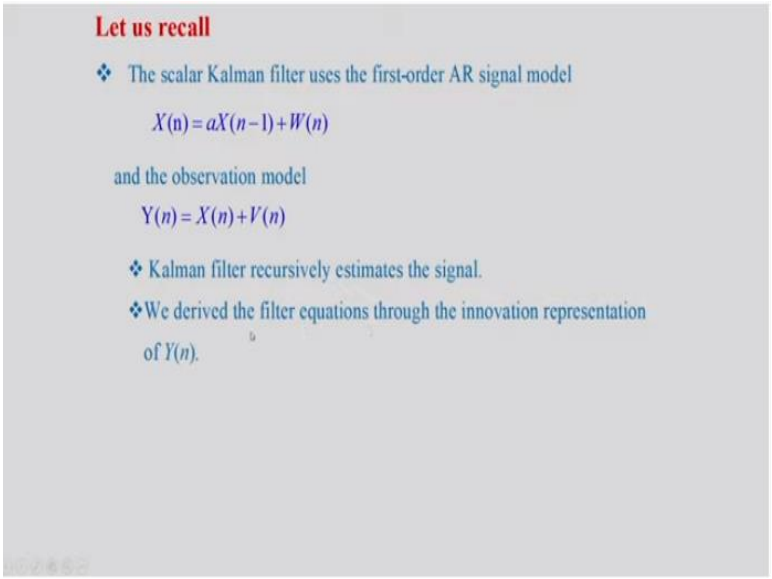
Statistical Signal Processing
Prof. Prabin Kumar Bora
Department of Electronics and Electrical Engineering
Indian Institute of Technology – Guwahati

Lecture – 36
Vector Kalman Filter

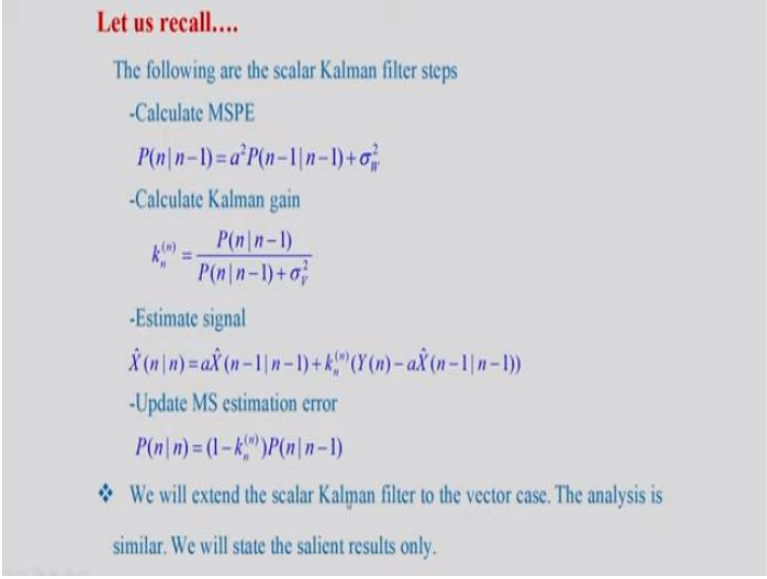
(Refer Slide Time: 00:44)



Hello students welcome to this lecture on vector Kalman filter. Let us recall the scalar Kalman filter uses the first order autoregressive signal model that is Xn = a times Xn – 1 + Wn where Wn is a white noise and is a constant to tackle non-stationarity. It can be function of n the observation model is given by Yn = Xn + Vn. Again Vn is a white noise uncorrelated with Xn and uncorrelated with Wn.

Kalman filter recursively estimates the signal. So, that we know that Kalman filter recursively estimate the signal on the basis of the current and all the previous data. We derived the filter equations true innovation representation of Yn for data Yn we had a innovation representation and then we derived the scalar Kalman filter.

(Refer Slide Time: 01:48)

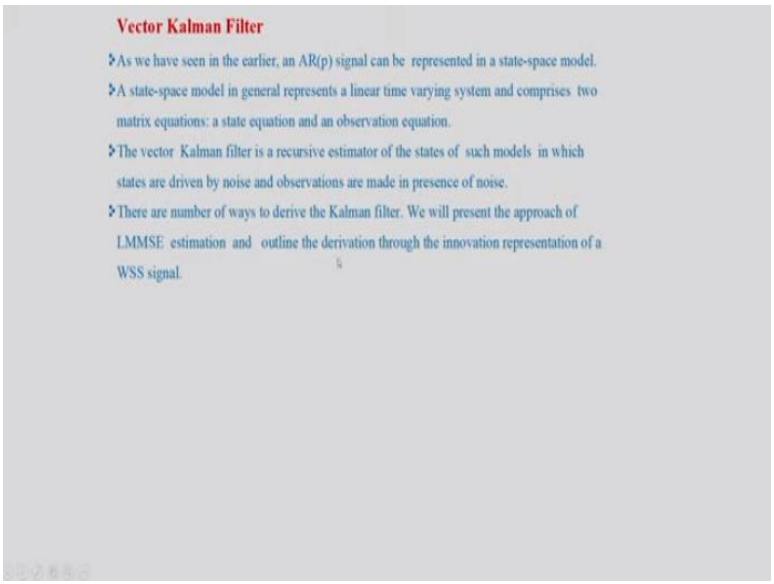


The following are the scalar Kalman filtering steps. First calculate the mean square prediction error this formula P of n given n - 1 = a square times P of n - 1 given n - 1 + sigma W square so this is the mean square error estimated at time n - 1 then we calculate the Kalman gain Kn n that equal to P of n given n - 1 that is the mean square prediction error divided by P of n given n - 1 means square prediction error + sigma V square.

Then we estimate the signal X of n given n = a times X at n - 1 given n - 1 this is the predicted part and there is a correction part here into Kn n that is Kalman gain multiplied by the innovation part Yn - a times X at n - 1 given n - 1 this is the estimator at instant n - 1. Then we update the MS estimation error P of n given n = 1 - K and n into P of n given n - 1. This is the step for updating the mean square error now we will extend this scalar Kalman filter to the vector case.

The analysis is similar we will state the salient results only we will not go into detail analysis, but this derivation is also based on the innovation representation of the data. Now we will introduce vector Kalman filter.

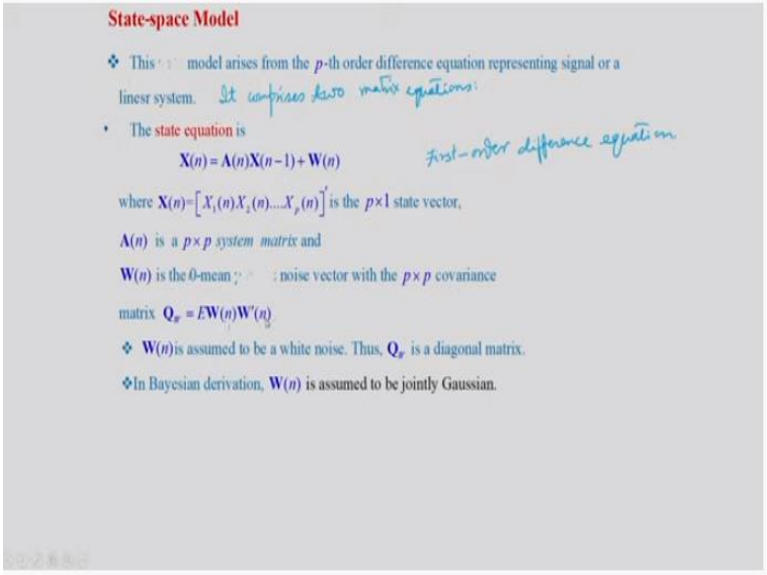
(Refer Slide Time: 03:53)



As we have seen in the earlier an ARP signal can be represented in a state space model. A state-space model in general represents a linear time varying system and comprises two matrix equations a state equation and an observation equation. This is the state space model used by the control engineers. The vector Kalman filter is a recursive estimator of the states of such model in which states are driven by noise we will solve the model is the state driven by noise and observations are made in the presence of noise.

So, that way we have noisy observations and we have to estimate the states. There are number of ways to derive the Kalman filter for example there is a Bayesian approach. We will present the applause of linear mean square error estimation and outline the derivation through the innovation representation of a WSS signal.

(Refer Slide Time: 05:04)



Let us discuss this state space model this model arises from the pth order difference equation representing signal or a linear system. For example, in the case of ARP signal autoregressive signal of order P, we have a pth order difference equation. For example, in the case of ARP model we have a pth order difference equation. Similarly, a discrete-time linear system can be described by a pth order difference equation.

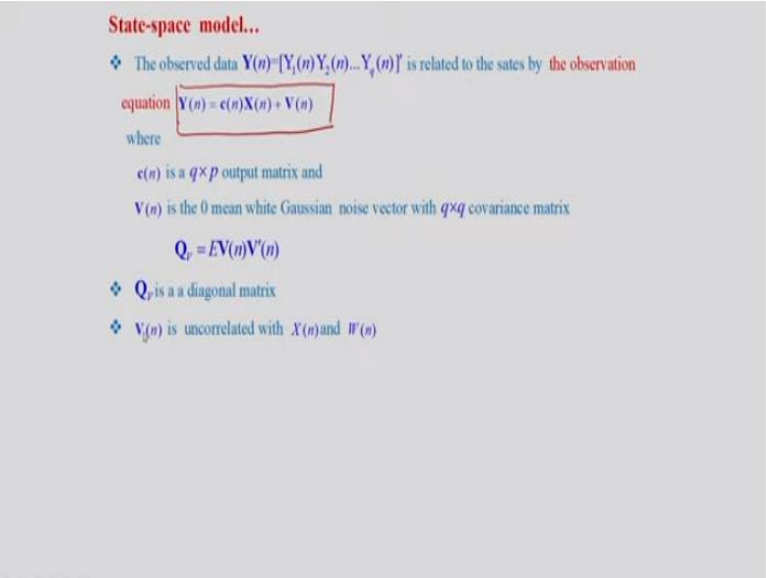
Now the state space model comprises of two situations it comprises two matrix equation actually matrix equations number one is state equation this state equation is given by this Xn = An into X n - 1 + Wn. So, here this is a first-order difference equation unlike in the case of general model where it was pth order differentiation this is first order difference equation. Now here Xn is the state vector comprises of P state variable x1 and x2 n up to Xp.

And so this is this state vector and An is a p/p because P dimensional state vector is there so An will be p/p system matrix or state matrix. Wn is a 0 mean noise vector with the p/p covariance matrix that is Qw = E of Wn into Wn transpose. This Wn it is a noise vector which is known also known as the process noise and it is state component will have corresponding noise addition and this noise is a white noise.

And it is characterized by the covariance matrix E of Wn into Wn and transpose since Wn is assumed to be a white noise QW is a diagonal matrix. So this QW matrix is a diagonal matrix. In

Bayesian derivation Wn is assumed to be jointly Gaussian also in addition to white this is assumed to be Gaussian also.

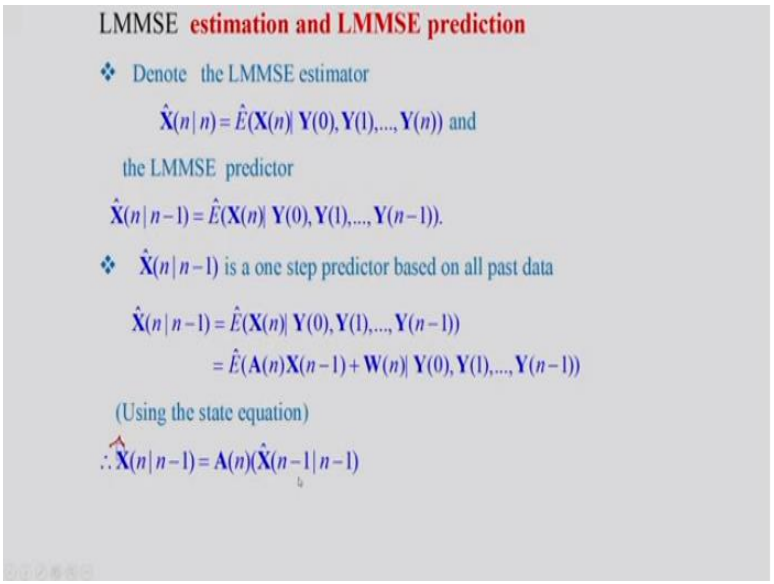
(Refer Slide Time: 07:47)



Now we will see the observation model the observed data Yn that is comprising of Y1 Y2 up to Y qn there are q observation at a instant n. So, this is represented in terms of vector and that is the Yn vector it is related to the states by the observation equation that is Yn = cn into Xn +Vn. So, this is the observation equation Yn = cn into Xn + Vn. So this is the observation equation now here cn is a q / p output matrix because this Yn is a Q dimensional vector therefore this matrix is q/p matrix.

Vn is a 0 mean white Gaussian noise vector this is their measurement error or measurement noise which is also assumed to be white Gaussian is necessary only in the basic derivation and its covariance matrix is a q / q covariance matrix which is given by Qv = EVn into Vn transpose Qv is a diagonal matrix because Vn is a white noise vector and further it is assumed that Vn is uncorrelated with both Xn and Wn. So, this Vn is uncorrelated with accent and also with the step noise that is Wn. So, therefore Vn is uncorrelated with Xn that is the state and the Wn that is the state noise or process noise.

(Refer Slide Time: 09:47)

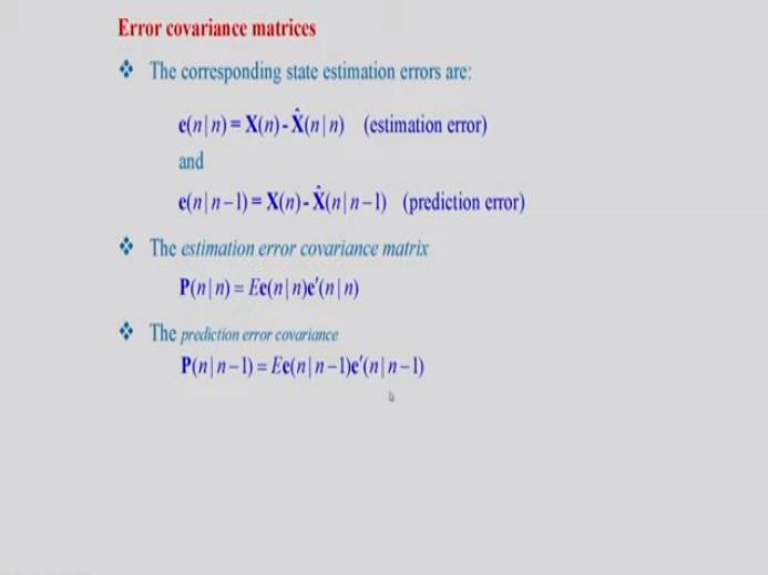


Now let us see linear minimum mean square error estimation and linear minimum mean square error prediction associated with Kalman filter. Denote the LMMSE estimator by this X hat n given n E hat is the symbol for linear minimum mean square error estimator. Therefore, X hat n given n is E hat of Xn given all the data up to present Y0, Y1 up to Yn and the linear minimum mean square error predictor is given by this X hat n given n - 1 data up to n - 1.

So, that way it is E hat of Xn given Y0, Y1, up to Yn - 1. So, therefore this is a one-step predictor based on all the past data. Now this quantity X hat n given n - 1 that we can write as E hat of Xn given Y0, Y1 up to Yn – 1. Now Xn, we can write like An into Xn – 1 + Wn. So, this is the state model we are using so using this state equation we will get the above expression = E hat of An into Xn – 1 + Wn given Y0, Y1 up to Yn - 1 and now this is E hat of Xn - 1 given Y0, Y1 up to Y n - 1.

Let LMMSE or linear minimum mean square error estimation of Xn - 1 given data up to n - 1 so that way it is X hat n - 1 given n - 1 therefore X hat given n - 1 is An times X hat n - 1 given n - 1. So, this prediction is related to the previous estimation by this relationship An times the previous estimation.

(Refer Slide Time: 12:09)

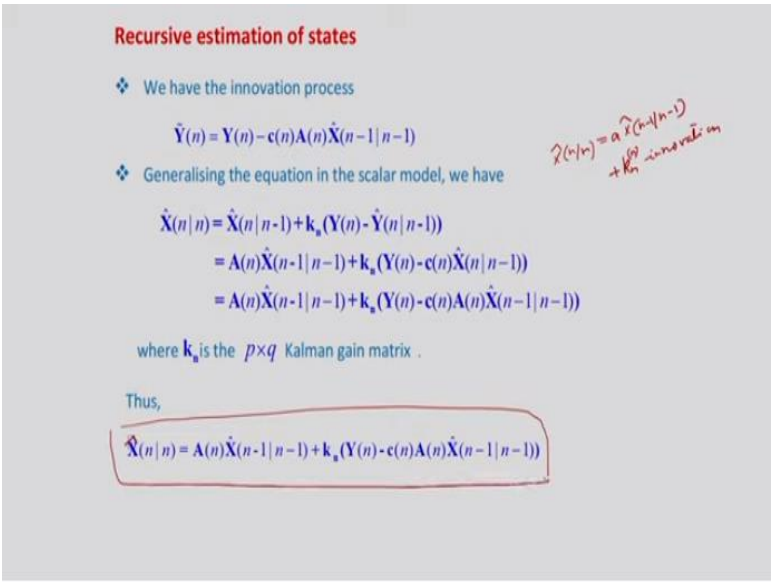


We can find out the corresponding error covariance matrices also suppose estimation error E of n given n = Xn – X hat n given n this is the estimator of Xn given data up to n. Similarly E of n given n - 1 that is Xn minus the prediction; that is the prediction of Xn given data up to n - 1. So, that way this is the prediction error this is the estimation error. Now we can define the error covariance matrix in scalar case we are interested only in mean square prediction error but here we will be considering the error covariance matrix.

So, that way P of n given n that is the notation for error covariance matrix. So, this is the estimation error covariance matrix and it is given by E of en given n that is the estimation error into en given n transpose. So, estimation error vector into estimation error vector transpose. So, this is the covariance matrix so estimation error covariance matrix. Similarly, prediction error covariance matrix we can find out that is denoted by P of n given n - 1 this is expected value of en given n - 1 into en given n - 1 transpose. This is the prediction error vector.

This is the transpose of the prediction error. So, that way we get the two covariance matrices that is P of n given n this is the estimation error covariance matrix and another covariance matrix is prediction error covariance matrix so that is given by this p of n given n - 1 = E of en given n - 1 into en given n - 1 transpose prediction error vector into prediction error vector transpose.

(Refer Slide Time: 14:33)



Now let us see the innovation representation for a vector case we know the innovation representation for this scalar case. So, the derivation of the Kalman filter is based on the innovation of Yn and which is given by this expression Y tilde and E = Yn -the linear estimator of Yn given data up to n - 1. So, e hat of Yn given Y0, Y1 up to Yn – 1; we have Yn; this is the Yn vector = cn matrix into state vector Xn vector + Vn, Vn is a white noise vector. So, this is the observation model that we are substituting here.

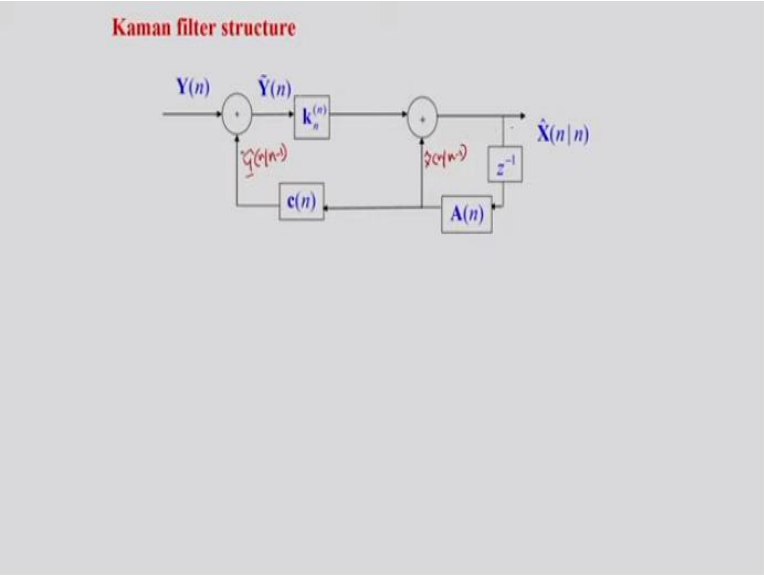
So, this will be E hat of Cn into Xn + Vn given Y0, Y1 up to Yn - 1 and this will be because now Vn this is the noise which is independent of all path data therefore this part will contribute 0 therefore we will have Yn - cn into X hat n given n - 1. Now this quantity also X hat n given n - 1 = An X hat n - 1 given n - 1. The prediction is obtained through the model that is An it into X hat n - 1 given n - 1 therefore Y tilde n will be equal to Yn – cn, cn matrix into An into X hat n - 1 given n - 1 this is the previous estimation.

Thus we have obtained the innovation of data that is given by Y tilde n = Yn - cn into An into X hat n - 1 given n - 1. So, this has come through the model we can use the innovation to get the recursive estimation of states we have y tilde = Yn - cn into An into X hat n - 1 given n - 1. Now generalizing this scalar model in scalar model we had X hat n given n = A times X hat n - 1 given n – 1 + Kn times Kn n times the innovation and here we omit this super fixed n.

So, simply we write Kn therefore X hat n in this case it will be a vector now X hat n given n = that predictor part that is X hat n given n – 1 plus a correction based on the innovation that is Kn x into Yn - Y hat n given n - 1 this is the prediction and this we can write as An into X hat n - 1 given n - 1 that is the estimation part at the instant n - 1 into Kn times Yn - and this we can write through the model cn into X hat n given n - 1.

Now we can write here also X hat given n – 1 = An into X hat n - 1 given n - 1. Therefore, our X hat n given n will be n into X hat n - 1 given n – 1 + Kn times Yn - cn into An into X hat n - 1 given n - 1 where this Kn, Kn is the p / q Kalman gain matrix. This is the matrix ways the innovation part thus we can write X hat n given n = An into X hat n - 1 given n – 1 + Kn times Yn - cn into An into X hat n - 1 given n - 1. So this is the state estimator for the Kalman filter from this expression we can have the Kalman filter structure.

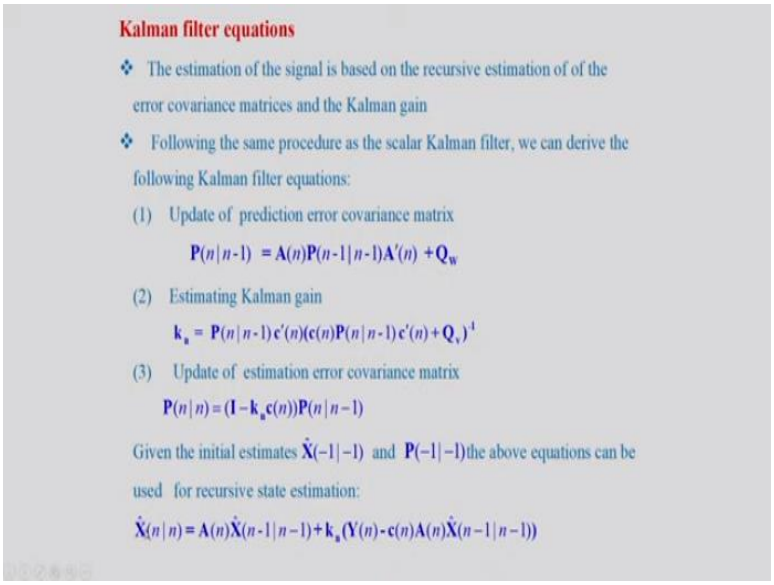
(Refer Slide Time: 19:56)



So, there is data here data is Yn and the state estimator is X hat n given n. Now this estimator is delayed by one unit and then multiplied by An to get the prediction X hat n given n - 1. So, we will get here X hat n given n - 1 given that up to n - 1 we will get the prediction below through the delayed version of this state estimator. So, when we will delay this, we will get X hat n - 1 given n - 1 that will be multiplied by An and we will get this prediction.

And this result is multiplied by cn matrix and then we will get the predicted Yn that is Y hat n given n - 1. So, this is the prediction of the data vector at instant n because of the data up to n - 1 so this is the prediction vector we can give a bar here to denote vector. So, that way this is the prediction vector this is the actual data and then difference is the innovation and that innovation is scaled by the Kalman gain and then edit with the predicted part to get the estimator. So, that way estimator has a prediction part here and then there is a collection part which is the innovation multiplied by the Kalman gain.

(Refer Slide Time: 21:43)



Now we will state the Kalman filter equation the estimation of the signal is based on the recursive estimation of the error covariance matrices and Kalman gain. So, we will see how they are estimated following this same procedure as the scalar Kalman filter we can derive the following Kalman filter equations. That is update of prediction error covariance matrix P of n given n – 1 = n times Pn - 1 given n - 1 into An transpose. An is the time varying system matrix + Qw, Qw is the covariance matrix of the process noise.

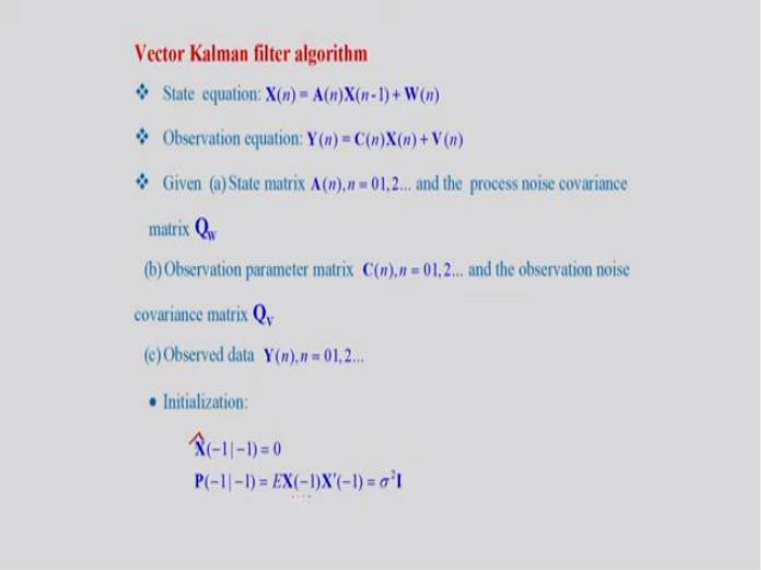
So, therefore the prediction error covariance is given by this P of n given n – 1 = An into P of n - 1 given n - 1 this is the estimation error covariance matrix multiplied by n transpose + Qw. First one is the update of the prediction error covariance matrix and this is given by P of n given n - 1= An into P of n - 1 given n - 1 into An transpose + qw where An is the system matrix or state matrix and QW is the covariance matrix of the noise vector.

So, that way we can find out the prediction error covariance in terms of the estimation error covariance of previous iteration. Then estimating the Kalman gain so for that we have Kn = P of n given n - 1 we should obtain here into cn transpose this is the output matrix transpose into cn into Pn given n - 1 into cn transpose + Qv whole inverse. So, this part is a matrix and inverse of this is taken.

So, that way the Kalman gain is given by the error covariance matrix multiplied by this output matrix transpose multiplied by this inverse, inverse of cn into Pn given n - 1 into cn transpose + Qv, Qv is the covariance matrix of the other observation noise. So, once we have the Kalman gain we can update the estimation error covariance P of n given n = I - Kn into cn this is the Kalman gain into cn into P of n given n – 1.

So, given the prediction error covariance we can get the estimation error covariance. However along with this equation we must have the initial estimate that is X hat of - 1 given - 1 and the estimation for P of - 1 given - 1 that is the covariance matrix of the mean square error one estimate must be there. Now given these initial estimates X hat - 1 given - 1 and P - 1 given - 1 the above equations these 3 equations can be used to recursively estimate the state and the recursive state estimator is given by this X hat n given n = An into X hat n - 1 given n – 1 + Kalman gain Kn into Yn - cn into An into X hat n - 1 given n – 1. So, this is the innovation part, and this is scaled by Kn and added to the predicted part.

(Refer Slide Time: 26:14)

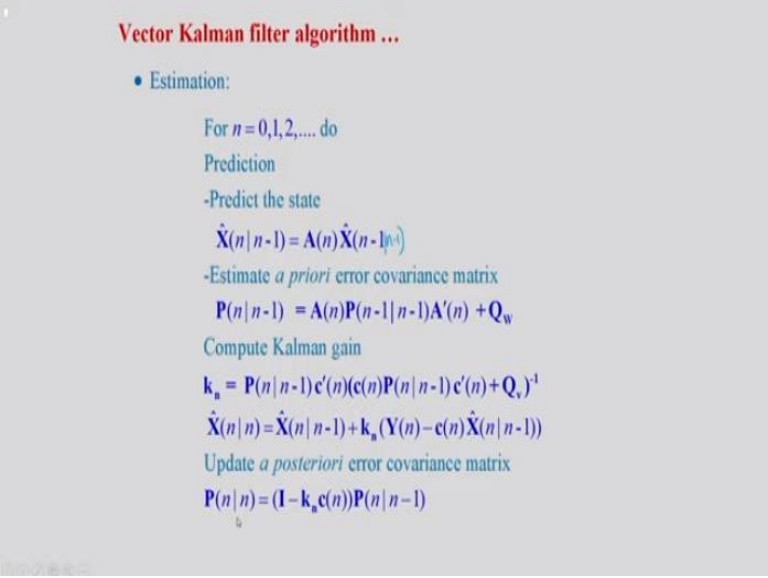


With these backgrounds we can now state the Kalman filter by algorithm. So, here we have to know this state equation Xn = An Xn – 1 + Wn observation equation that is given by Yn = cn into Xn + Vn this is the observation equation. So, we must be given a state matrix An for n = 0 to P because it is pth order system or signal, and the process noise covariance matrix is given by Qw which is a diagonal matrix.

The observation parameter matrix that is cn for n = 0, 1 up to 2 if it is time varying otherwise it will be a simply C matrix and the observation noise covariance matrix is given by Qv this is again a diagonal matrix and observed data Yn that is from n = 0, 1, 2 etc. So, these are the things needed for the algorithm now it will be initialized as X hat - 1 given – 1 = 0 all states are initialized with 0 and this estimation error covariance matrix at instant – 1.

There is no estimation therefore this will be the covariance of the data itself people would know this statistical model we can use that and here we can write it as sigma square into I. This is the estimator and if we take this type of initialization sigma square into I where I is an identity matrix then the Kalman equations converges. So, that way this sigma square is a positive number which can be based on the estimation of this quantity or we can assume some value.

(Refer Slide Time: 28:25)



Kalman filter runs like this for n = 0, 1, 2 etc first we will predict so predict this state that is X hat n given n - 1 that is given by An into a X hat n - 1 given n – 1. So this is the estimated value this predicted term will be used in the updating here then we will have to estimate a priori error covariance matrix or prediction error covariance matrix that is given where P of n given n - 1 and this is related to previous estimation error covariance matrix by this relationship.

This is the state matrix An into Pn - 1 given n - 1 into n transpose + Qw, Qw is the covariance matrix the process noise so we have the a priori error covariance matrix or prediction error covariance matrix. Now we will complete the Kalman gain that is given by this matrix Kn matrix that is related to the error covariance matrix P of n given n - 1 multiplied by cn transpose into inverse of this quantity cn + pn given n - 1 into cn transpose + Qv whole inverse.

So, this is the expression for Kalman gain now once we have the Kalman gain we can now update this step that is X hat n given n = X hat n given n - 1 which we have obtained here and then Kn times the prediction error Yn - cn into X hat n given n - 1 this is the innovation or the prediction error. Now after this we have to update the a posteriori error covariance matrix that is given by P of n given n = I - Kn into cn into P of n given n - 1.

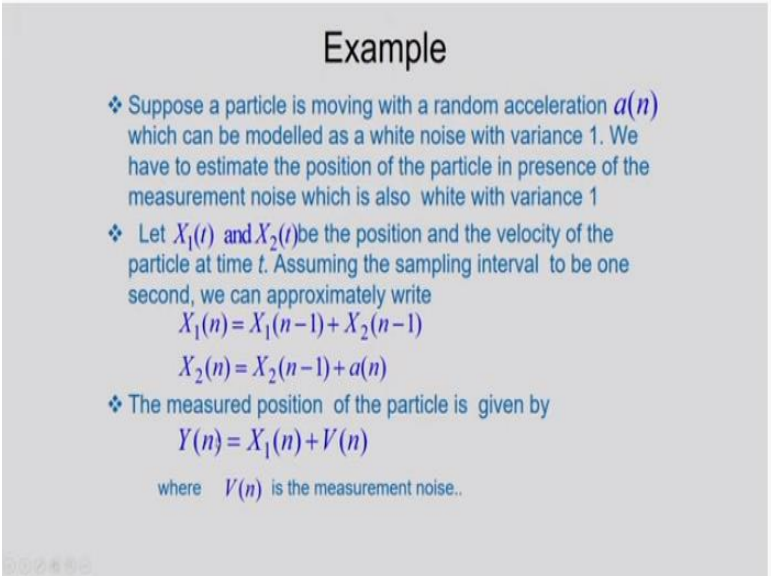
This is the a priori error covariance matrix and this is pre multiplied by this vector to get the estimation error covariance matrix or a posteriorly error covariance matrix. So, this is the Kalman filter algorithm.

(Refer Slide Time: 31:16)

It is a simple algorithm, but it is widely used some applications are as follows Kalman filter has become an integral component of thousands of military and civilian navigation and guidance systems. So, any navigation or guidance system Kalman filter is a part. NASA adopted the Kalman filter to solve the problems associated with determining satellite orbits guided the Apollo 11 lunar module to the moon surface using this Kalman filter.

The Global Positioning System GPS is based on this deceptively simple algorithm. Similarly, it can be used for video based tracking time series based forecasting of financial data etc. There are many applications and there are many improved version of this algorithm.

(Refer Slide Time: 32:21)

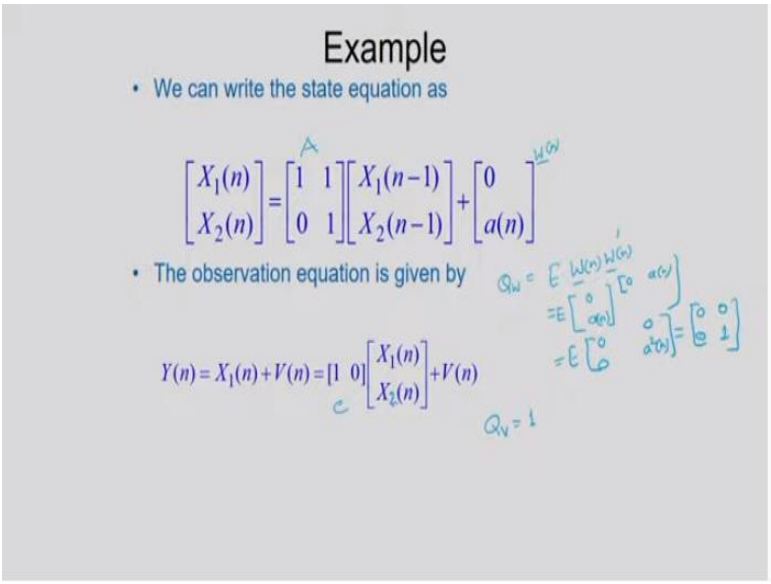


Let us consider one example suppose a particle is moving with a random acceleration an which can be modeled as a white noise with variance 1. Acceleration is random with variance 1 we have to estimate the position of the particle in presence of the measurement noise which is also white with variance 1. Let us solve this problem let X1 t and X2 t be the position and the velocity of the particle at time.

These two will be the state variables assuming this sampling interval to be one second, we can approximately write that is position X1n at time n = X1n - 1 that is the position at in time n – 1 plus distance traveled in one second that is X2 n - 1 that is the velocity multiplied by 1. Therefore, X1n = X1n – 1 + X2 n - 1 and velocity at instant and X2n = X2 n - 1 velocity at instant – 1 plus the acceleration into 1.

Because after one second, we are considering so that way this part we can write as X2 n = X2 n – 1 + an. The measured position of the particle is given by Yn = X1 + Vn this is the observed variable that is observed data Yn = X1n + Vn where Vn is a measurement noise which is given to be white with variance 1.

(Refer Slide Time: 34:30)



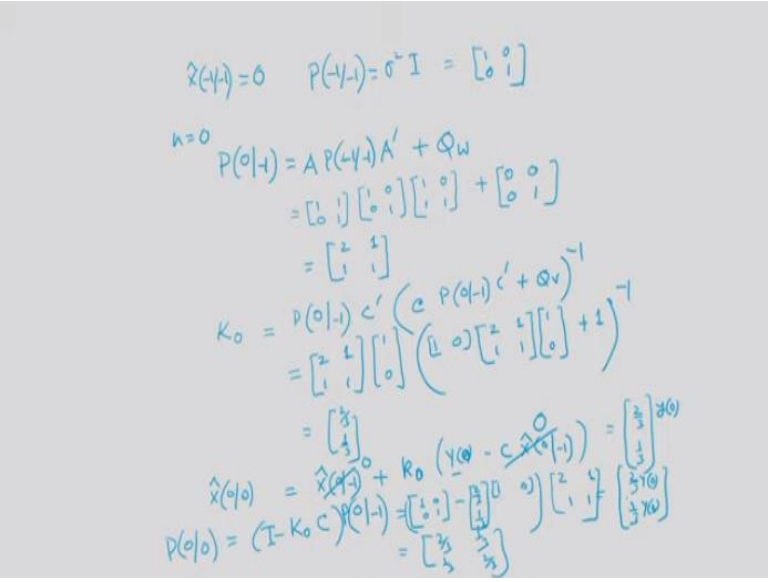
So, we can write from these expressions X1n = X1n – 1 + X2 n- 1 and X2n = X2 n- 1 + an we can write the matrix states equations X1 and X2n these are the states = 1 1 0 1 that is the system matrix multiplied by X1 n - 1 X2 n- 1 these are the previous states plus first equation we do not have any noise term. So, that is 0 and second equation we have a noise term an so this is the term corresponding to Wn.

So, this is the state equation the observation equation is given by Yn = X1n + Vn because position plus noise and this we can write as because it is X1n. So, we can write it in terms of a matrix that is there 1 0 matrix this is a C matrix multiplied by X1n X2n. So, these are the states if I multiply it 1 0 that is the C matrix into state vector X1n X2n then 1 into X1n + 0 into X2n and we will get X1n and + Vn.

So, Vn is the noise and it will have a variance because only the single dimensional quantities it will have variance which is sigma v square which is equal to 1. Thus here we can write that Qw that is a matrix because it is a of this if this is w and vector Wn into Wn transpose. So, that will be equal to E of 0 an transpose will be 0 an and we will get E of 0 and this will be 0 and this one will be 0 and here a square and if we take the expected value, we will get 1 here because variance is 1.

So, that way it is 0 0 0 1 so this is the error covariance matrix here, so we know Qw we know this is my a which is constant not time varying this is my C matrix and you Qv is now it one-dimensional quantity that is equal to sigma V square actually that is equal to 1. So, these are the parameters given with this we can estimate this state.

(Refer Slide Time: 37:54)



So, initial value we can assume that X hat - 1 given - 1 that is initially this = 0 and then P of - 1 that is error covariance matrix given - 1 that will be also equal to some positive matrix we have to consider so that where sigma square I where sigma square is a positive quantity. We may take this is sigma square make we may take one so we can take this as the identity matrix itself. So, this is 1 0 0 1 now starting with n = 0 we will first estimate the prediction error covariance matrix so that way we will estimate P of 0 given – 1.

So, this is given by A into P - 1 given - 1 this is the initial estimate of the error covariance matrix into A transpose + Qw this is the expression. So, this will obtain from this expression An into Pn - 1 given n - 1 into An transpose + Qw. So, we get this expression and we can carry out the computation now this = 1 1 0 1 this is my A and then this is identity matrix 1 0 0 1 and this is A transpose that way it will be 1 0 1 1 + Qw, Qw matrix we have found out Qw matrix is 0 0 0 1.

So, that way it will be 0 0 0 0 0 0 1 so we can carry out this computation and we will get this as 2 1 1 1 so this is the a priori error covariance matrix. Now when this we can find out K0 Kalman

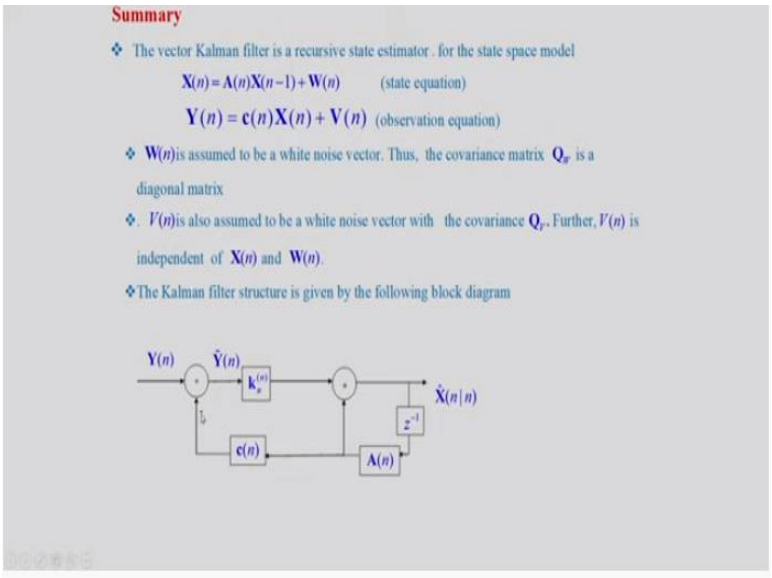
gain at time 0 = P of 0 given - 1 into c transpose into c into P of 0 given - 1 into c transpose + Qv whole inverse. So, this is the expression we get from this formula, so this is the Kalman gain. So, using this formula we get this, so we have already obtained this, and we have value of c, c is 1 0 here c is known this one is known here c transpose we know Qv is simply 1.

So, that way we can write this as this is this matrix 2 1 1 1 into c transpose is c is 1 0. So, this will be 1 0 and here similarly this will be 1 0 this is the matrix then this is 2 1 1 1 and then c transpose will be 1 0 + Qv = 1 and whole inverse so this we can carry out and we will get the here this matrix this is a 2 / 1 matrix two third and one third. Now we can estimate x0 given 0 given 0 that is will be equal to x0 given – 1. So plus Kalman gain that is k0 matrix into Yn - c times x0 given that is y0 here x0 given – 1.

So, that way we can write, and this part is 0 because there is no prediction initially this part is also equal to 0 and therefore, we will simply have ko, ko is two third one third into y0. So, that way this will be two third of y0 in and one-third of y0 so this is the estimator for the states this is the position this is the velocity and now we can calculate P of 0 given 0 that will be equal to I – I had to be identity matrix into k0 Kalman gain matrix into c all are given into P of 0 given – 1.

So, we know all this quantity, so this is the identity matrix 1 0 0 1 and then k0 we know that is equal to two third one third this matrix then c we know c is 1 0 and then this P0 given - 1 that also we know that is given by 2 1 1 1. So, if we carry out this calculation, we will get this as the two third, one third, one third, two third so that way we can find out P0 0 then we can find out P1 given 0 then k1 X hat 1 given 1 like that. So, if we can estimate the state in this recursive manner.

(Refer Slide Time: 45:21)

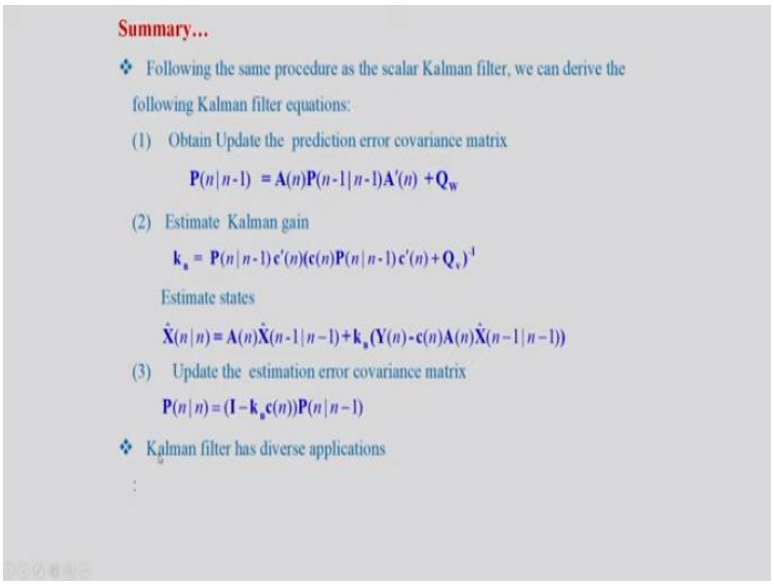


Let us summarize the lecture the vector Kalman filter is a recursive state estimator for the state space model this is this state space model Xn = An into Xn – 1 + Wn An is the state matrix or system matrix Xn is the state vector Wn is the process noise vector Yn that is the measurement vector or output data vector = cn that is known as the output matrix multiplied by the state vector + Vn, Vn is the measurement noise vector.

Wn is assumed to be white noise vector thus the covariance matrix Qw is a diagonal matrix Vn is also assumed to be a white noise vector with the covariance Qv. Further Vn is independent of Xn and Wn that is important the Kalman filter structure is given by the following block diagram. So, this is the input data Yn and this is the X hat n given n this is the output state estimator. So, this is the input that is the data vector.

Now this is delayed that will give us X hat n - 1 given n - 1 that will be multiplied by An and we will get the predicted part here and this predicted part is multiplied by cn to get the predicted data here we get the predicted state here, we get the predicted data then this is subtracted we get the innovation. This innovation we are scaling by the Kalman matrix and this is added to the predicted part to get the state estimator.

(Refer Slide Time: 47:30)



Following this same procedure at this scalar Kalman filter we can derive the following Kalman filter equations that is obtain the update of the prediction error or a priori error covariance matrix P of n given n - 1= An into P of n - 1 given n - 1 into n transpose + Qw. Estimate the Kalman gain that is Kn = P of n given n - 1 into cn transpose into inverse of this that is cn into Pn given n - 1 into cn transpose + Qv where Qv is the observation noise.

This is the covariance matrix of the observation noise then we will estimate these states that is X hat n given n = An into X hat n - 1 given n - 1 this is the prediction part + Kn Kalman gain into one correction part that is Yn – cn into An into X hat n - 1 given n – 1. This is the previous estimation now we will update the estimation error because here estimation error is in use. So, for that we have to update it and this update is given by I - Kn into cn into P of n given n - 1 this is the covariance matrix update equation. So, this is the Kalman filter algorithm and we are also stated that Kalman filter has diverse applications. Thank you.