Lecture Summary: Distributed Estimation

Vijay Gupta* and Richard M. Murray †

Static Sensor Fusion: The static sensor fusion problem is to obtain the best linear minimum mean squared error (MMSE) estimate of a random variable X given multiple measurements $y_i = H_i x + v_i$, at sensors $i = 1, 2, \cdots, n$. Assume that the noises v_i are mutually uncorrelated, zero mean, with covariance $R_{v,i}$, and independent of the zero mean random variable X with covariance R_x. One obvious solution is for every sensor to transmit its measurement to a central station that computes the MMSE estimate. However, solutions that place some of the computational load on the sensors may be preferable.

Basics: Let $y = Hx + v$, where H is a matrix and v is a zero mean Gaussian noise with covariance R_v independent of the zero mean random variable X with covariance R_x. The best linear minimum mean squared error (MMSE) estimate of X given $Y = y$ is given by $\hat{x} = R_x H^T \left(HR_x H^T + R_v\right)^{-1} y$, with the error covariance $P = R_x - R_x H^T \left(HR_x H^T + R_v\right)^{-1} HR_x$. Using the matrix inversion formula, this can be re-written as $P^{-1}\hat{x} = H^T R_v^{-1} y$, where $P^{-1} = R_x^{-1} + H^T R_v^{-1} H$.

Star Topology: This result can be used to solve the static sensor fusion problem when a central station can receive information from all the sensors. The sensors calculate local estimates $\hat{x}_i$ of x based on local measurement y_i. These local estimates and the corresponding error covariances P_i are transmitted to the central station. The central station can compute the global estimate based on all the measurements as $P^{-1}\hat{x} = \sum_{i=1}^{n} P_i^{-1}\hat{x}_i$, where P is the associated global error covariance given by $P^{-1} = \sum_{i=1}^{n} P_i^{-1} - (n-1)R_x^{-1}$. This can be proven by utilizing the fact that the noises v_i are mutually uncorrelated.

Arbitrary Topology: The situation when the sensors are connected in an arbitrary topology, and every sensor requires the global estimate, is more difficult. One possibility is to use average consensus-like algorithms to calculate weighted average of vector values held by the nodes. Every node i, sets an initial value $x_i(0) = \hat{x}_i$, and executes the iteration

$$x_i(k+1) = x_i(k) + hP_i \sum j : j \text{ is a neighbor of } i(x_i(k) - x_j(k)),$$

then the nodes converge asymptotically to the global estimate.

Dynamic Sensor Fusion: The dynamic sensor fusion problem is similar to the static sensor fusion problem, except that the underlying variable that needs to be estimated evolves in time as

$$x(k+1) = Ax(k) + w(k),$$

where $w(k)$ is process noise, usually modeled as white, Gaussian, zero mean, and with covariance R_w. Each sensor i generates a local measurement at every time k according to

$$y_i(k) = Cx(k) + v_i(k),$$

where measurement noises $v_i(k)$ are white, Gaussian, zero mean, with covariance $R_{v,i}$, mutually uncorrelated, and independent of the process noise. The initial condition $x(0)$, is Gaussian with zero mean, covariance $P(0)$, and independent of all noises. The estimate $\hat{x}(k)$ of state $x(k)$ needs to be calculated based on measurements from all sensors till time k.

*Department of Electrical Engineering, University of Notre Dame, vgupta2@nd.edu
†Control and Dynamical Systems, California Institute of Technology murray@caltech.edu

Transmitting local estimates is not enough: Unlike the static case, in general, the global estimate (based on measurements from all the sensors), cannot be generated from local estimates (based on local measurements). The basic reason is that the process noise leads to the estimate errors being correlated. The general result is as follows. Suppose two sets of measurements Y_1 and Y_2 are used to generate local estimates $\hat{x}_1$ and $\hat{x}_2$. Let

$$\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = L \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} \doteq LY.$$

Then the global estimate $\hat{x}$ can be calculated from $\hat{x}_1$ and $\hat{x}_2$ if and only if

$$R_{YY}L^T \left(LR_{YY}L^T\right)^{-1} LR_{YX} = R_{YX}.$$

For the dynamic sensor fusion case, Y_i can be interpreted as measurements from i-th sensor till time k. Thus, as k progresses, L would be a fat matrix and thus condition will not be satisfied. Thus, the interest shifts to obtaining the quantities that need to be transmitted by the sensors (two-block problem) to ensure that the global estimate can be calculated.

Information form of the Kalman filter: Consider a random variable evolving in time as

$$x(k+1) = Ax(k) + w(k).$$

Suppose it is observed through measurements of the form

$$y(k) = Cx(k) + v(k).$$

Then the measurement updates of the Kalman filter can be given by the alternate information form:

$$P^{-1}(k|k)\hat{x}(k|k) = P^{-1}(k|k-1)\hat{x}(k|k-1) + C^T R^{-1} y(k)$$
$$P^{-1}(k|k) = P^{-1}(k|k-1) + C^T R^{-1} C$$

Distributed Kalman Filtering: The global estimate and error covariance matrix can be calculated in terms of local estimates and error covariance matrices as

$$P^{-1}(k|k) = P^{-1}(k|k-1) + \sum_{i=1}^{n} \left(P_i^{-1}(k|k) - P^{-1}(k|k-1)\right)$$
$$P^{-1}(k|k)\hat{x}(k|k) = P^{-1}(k|k-1)\hat{x}(k|k-1) + \sum_{i=1}^{n} \left(P_i^{-1}(k|k)\hat{x}_i(k|k) - P_i^{-1}(k|k-1)\hat{x}_i(k|k-1)\right).$$

This suggests two architectures for distributed Kalman filtering (and hence dynamic sensor fusion). Note that both the architectures assume a star topology with a central station communicating with every sensor.

- The individual sensors transmit $\hat{x}_i(k|k)$ at every time step. The global estimate is calculated using the above result. This strategy assumes that communication occurs at every time step for data fusion. The time update steps do not require any communication.
- The i-th sensor calculates a term $z_i(k)$ that evolves as

$$z_i(k+1) = P^{-1}(k|k-1)AP(k-1|k-1)z_i(k) + \left(P_i^{-1}(k-1|k-1)\hat{x}_i(k-1|k-1) - P_i^{-1}(k-1|k-2)\hat{x}_i(k-1|k-2)\right).$$

  The global error covariance matrices can be calculated by each sensor if it knows what sensors have communicated at every time so far. The estimator calculates

$$P^{-1}(k|k-1)\hat{x}(k|k-1) = \sum_{i=1}^{n} z_i(k).$$

If the sensors are connected using an arbitrary topology, the general problem is still open. One possibility is to use several iterations of an average consensus like algorithm for every measurement update and fusion step using the distributed Kalman filtering equations stated above. Finally, note that the second architecture can be used even if sensors do not communicate at every time step (track-to-track fusion problem).