KALMAN FILTER:

Dynamics: $x(k+1) = Ax(k) + w(k)$ $w(k) \sim N(0, W)$

Sensor: $z(k) = Hx(k) + v(k)$ $v(k) \sim N(0, V)$

Measurement

FILTER: $\hat{x}(k)$, $\Sigma(k)$

state estimate

covariance estimate

① PREDICTION:
$$\hat{x}(k|k-1) = A\hat{x}(k-1)$$
$$\Sigma(k|k-1) = A\Sigma(k-1)A^T + W$$

min

② MEASUREMENT:
$$\hat{x}(k) = \hat{x}(k|k-1) + K(k)\left(z(k) - H\hat{x}(k|k-1)\right)$$

optimal gain to shrink covariance

sensor output

expect to see

ea. node. AP

MS

$$\Sigma(k) = (I - K(k)H)\Sigma(k|k-1)$$

GAIN: $K(k) = \Sigma(k|k-1)H^T\left(H^T\Sigma(k|k-1)H + V\right)^{-1}$
$$= \Sigma(k)H^T V^{-1}$$

INFORMATION FILTER:

NEW VARIABLES: $\hat{y}(k) = I(k)\hat{x}(k)$ $I(k) = \Sigma(k)^{-1}$

$$\Sigma(k) = \Sigma(k|k-1) - \Sigma(k|k-1)H^T\left(H^T\Sigma(k|k-1)H + V\right)^{-1}H\,\Sigma(k|k-1)$$

$$\Sigma^+ = \Sigma - \Sigma H^T\left(H^T\Sigma H + V\right)^{-1}H\Sigma$$

$(A+B)^{-1} = ?$ (if we know A^{-1} ...)

WoodBury Matrix Identity:

$$(A + \underbrace{UCV}_{B})^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}$$

$\sim A^{-1}$

if low rank $\Rightarrow$ useful

$$\left([C]^{-1} + [\ V\ ][A^{-1}][U] \right)^{-1}$$

$$[A] + [U][C][\ V\ ]$$

if $U \in \mathbb{R}^{n\times 1}$ $V \in \mathbb{R}^{1\times n}$ $C \in \mathbb{R}$

called Sherman Morrison formula

Measurement

$$\Sigma^{+} = I^{-1} - I^{-1}H^{T}(H^{T}I^{-1}H+V)^{-1}HI^{-1} = (I + HV^{-1}H^{T})^{-1}$$

$$(\Sigma^{+})^{-1} = \boxed{\begin{aligned} I^{+} &= I + HV^{-1}H^{T} \\ \hat{y}^{+} &= \hat{y} + H^{T}V^{-1}z \end{aligned}}$$

clean (additive!)

Update to Dynamics ($\bar{\omega}$ information variables)

$$I(k|k-1) = L(k)M(k)L(k)^{T} + C(k)W^{-1}C(k)$$

$$\hat{y}(k|k-1) = L(k)A^{-T}\hat{y}(k-1)$$

- $M(k) = A^{-T}I(k)A^{-1}$
- $C(k) = M(k)(M(k) + W^{-1})^{-1}$

pretty ugly

- $L(k) = I - C(k)$

Distributed Version:

$$\begin{bmatrix} z_1(k) \\ \vdots \\ z_n(k) \end{bmatrix} = \underbrace{\begin{bmatrix} H_1(k) \\ \vdots \\ H_n(k) \end{bmatrix}}_{H} x(k) + \begin{bmatrix} v_1(k) \\ \vdots \\ v_n(k) \end{bmatrix} \qquad v_i \sim N(0, V_i)$$

$$I(k) = I(k|k-1) + \underbrace{\sum_{i=1}^{n} H_i(k)^T V_i^{-1} H_i(k)}_{H^T V^{-1} H}$$

$$\hat{y}(k) = \hat{y}(k|k-1) + \sum_{i=1}^{n} H_i^T(k) V_i^{-1} z_i$$

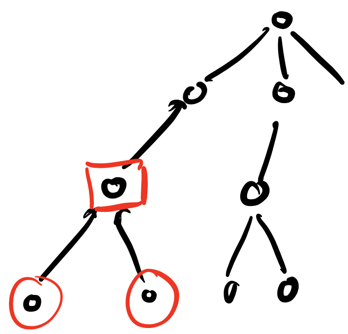
Distributed Scheme:

- ea. node keeps a copy of information matrix & state estimate

$$I_i(k) = I_i(k|k-1) + H_i(k)^T V_i^{-1} H_i(k)$$
$$\hat{y}_i(k) = \hat{y}_i(k|k-1) + H_i(k)^T V_i^{-1} z_i$$

- Option 1: full connected network
- ea. node passes $I_i, \hat{y}_i$ to other nodes.
- ea. sums up $I_i, \hat{y}_i$
- performs filter step:

- Option 2: Leader node / collector.
- network architecture that channels I_i, $\hat{y}_i$ to the collector node.
- partial sums can be done along the way.
- collector node performs prediction step using dynamics.



Possible Extensions

$$\hat{x}_i(k) = \hat{x}(k|k-1) + K_i^o\left(z_i(k) - H_i(k)\,\hat{x}_i(k|k-1)\right) + \sum_{j \in N(i)} K_{ij}^c\left(\hat{x}_j(k|k-1) - \hat{x}_i(k|k-1)\right)$$

lose guarantees of optimality.

KF Extensions

- Extended KF → nonlin. dynamics → $\hat{x}$
  linearized dynamics → Σ
- Unscented KF → nonlin dynamics → $\hat{x}$
- Particle filters
- Particle filters + KF } → Rao Blackwellized.] *

• feedback particle filter

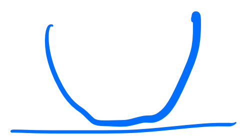
filtering... the end.

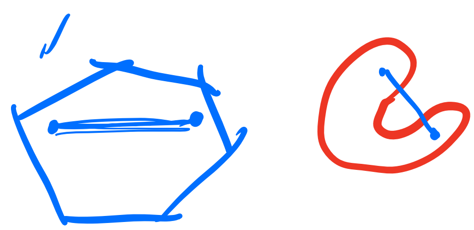
Distributed Optimization

$$\min_{x \in X} \underline{f(x)}$$ → objective function

$\underline{X}$ constrained set.

well behaved optimization ⟹ convex optimization
- convex objective
- convex constraints





dynamics → linear

optimization → convex

Aside:

Game: $\left\{ \min_{x_i} f_i(x) \right\}$ → ? Shahriar

Convex OPT: Rockafellar (ww) Convex Analysis ★
compact, flawless, unreadable.

Stephen Boyd → for engineers.

Distributed Optimization

$$\min_x f(x) = \sum_i f_i(x)$$

ea. agent node: knows about $f_i(x)$
and tracks their own estimate for $x \longrightarrow \hat{x}_i$

naively: ea. agent would do some gradient descent on $f_i(x)$ to update $\hat{x}_i$

Ex: from estimation: $f_i(x) = \| z_i - H_i x \|^2$

Dual Averaging: $\{ x(t), z(t) \}_{t=0}^{\infty}$ $\quad x \in X$, $z \in \mathbb{R}^d$

optimization (primal)

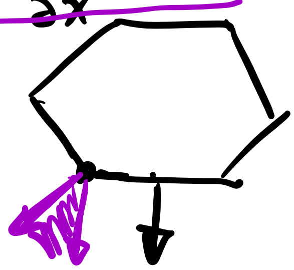
descent directions (dual)

$$\frac{1}{T} z(t+1) = \overset{0}{\cancel{z(t)}} - g(t) \qquad x(t+1) = \Pi_X^{\psi}\left(-z(t+1), \alpha(t)\right)$$

stepsize

$$g(t) \in \partial f(x(t))$$

$$\boxed{g(t) = \frac{\partial f}{\partial x}(x(t))}$$



$$\Pi_X^{\psi}(z, \alpha) := \underset{x \in X}{\operatorname{argmin}} \left[\langle z, x \rangle + \frac{1}{\alpha} \psi(x) \right]$$

$\psi(x)$: convex function

$$\psi(x) = \frac{1}{2} |x|_2^2$$

$$\psi(x) = \sum x_i \log x_i - x_i$$

if $x \in \mathbb{R}^n$ $\quad \underset{x}{\operatorname{argmin}} \left(\langle z, x \rangle + \frac{1}{\alpha} \frac{1}{2} \|x\|_2^2 \right)$

$$z^T + \alpha x^T = 0 \Rightarrow x = -\alpha z$$