

- Realistic sensor networks:** Normal nodes, faulty or corrupted nodes, malicious nodes
- Hierarchical scheme** – inspired from social and economic architectures which provides global trust on a particular context without requiring direct trust on the same context between all agents.
- Combines techniques from fusion centric, collaborative filtering, estimation propagation
- Trusted Core(TC)**
 - Trust Particles, higher security, additional sensing capabilities, broader observation of the system, confidentiality and integrity, multipath communication.
 - Every sensor can communicate with one or more trust particles by paying an communication and crypto cost.

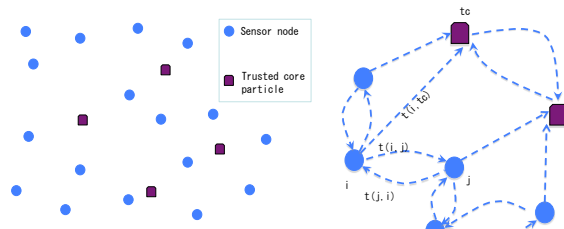


Figure 1: Sensor Network Realization

Figure 2: Local Dynamic Trust Relations

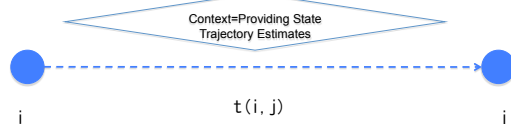


Figure 3: Context in trust relation

Goals of Trusted Core System Design

- All the sensors which abide by the protocols of sensing and message passing, should be able to track the trajectories.
- This implies that those nodes which have poor sensing capabilities, *nodes with corrupted sensors*, should be aided by their neighbors in tracking.
- Those nodes which are *malicious and pass false estimates*, should be quickly detected by the trust mechanism and their estimates should be discarded.

Gauss Markov System Model

$$\underline{x}[n + 1] = \mathbf{A}\underline{x}[n] + \mathbf{B}\underline{w}[n]$$

Observations

$$\underline{z}_i[n] = \mathbf{H}_i[n]\underline{x}[n] + \underline{v}_i[n]$$

Distributed Estimation Algorithms

Algorithm 1 Trusted Kalman Filter

```

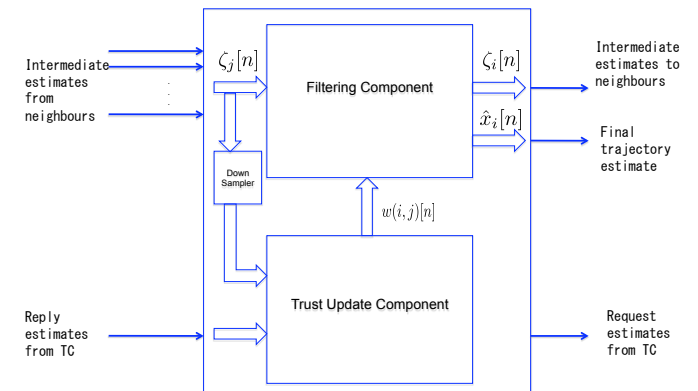
Init  $M[0], \hat{x}_i = \underline{x}(0), n = 0$ 
repeat
   $n \leftarrow n + 1;$ 
  Prediction MSE
   $P[n] = \mathbf{A}M[n-1]\mathbf{A}^T + \mathbf{B}Q\mathbf{B}^T$ 
  Kalman Gain
   $\mathbf{K}[n] = P[n]\mathbf{H}_i^T(\mathbf{R}_i + \mathbf{H}_iP[n]\mathbf{H}_i^T)^{-1}$ 
  Local correction
   $\zeta_i[n] = \mathbf{A}\hat{x}_i[n-1] + \mathbf{K}[n](\underline{z}_i[n] - \mathbf{H}_i\mathbf{A}\hat{x}_i[n-1])$ 
  The nodes exchanges the local estimates  $\hat{x}_j, \forall j \in \mathcal{N}^+(i)$ 
  Trust sensitive filtering
   $\hat{x}_i[n] = \sum_{j \in \mathcal{N}^+(i)} w_{ij} \times \zeta_j[n]$ 
  Estimation MSE
   $M[n] = (\mathbf{I} - \mathbf{K}[n]\mathbf{H}_i[n])P[n]$ 
until Forever
  
```

Algorithm 2 Trust Update for the inclusive neighborhood

```

Init  $t(i,j)[0] = \frac{1}{|\mathcal{N}^+(i)|}, \forall j \in \mathcal{N}^+(i),$  and  $k = 0$ 
repeat
  Wait for Exponential time  $\tau$ 
   $k \leftarrow k + \tau$ 
  Request Estimate update from the TC
  The TC replies with its trustworthy estimate  $\hat{x}_{tc}$ 
  for all  $j \in \mathcal{N}^+(i)$  do
     $dev(j) = \|\zeta_j - \zeta_{tc}\|_2$ 
     $t(i,j)[k] = \begin{cases} \min(\max_T, t(i,j)[k-1]) + \delta & dev(j) \leq Dev_T \\ t(i,j)[k-1]/2 & dev(j) > Dev_T \end{cases}$ 
  end for
until Forever
  
```

Component Architecture for Filtering Particles

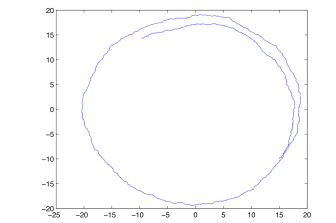


Simulations

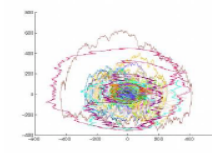
$$\mathbf{A} = 2 \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

$$\mathbf{Q} = 25\mathbf{I}_2, \quad \underline{x}(0) = (15, -10)^T$$

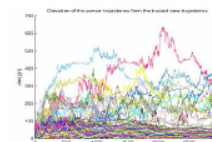
$$\mathbf{H}_{tc} = \mathbf{I}_2, \quad \mathbf{R}_{tc} = 30\mathbf{I}_2$$



System Orbit

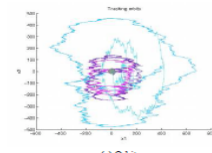


(a) Orbits

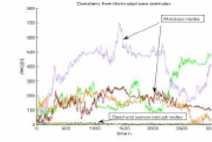


(b) Deviations

Noisy Open Loop Tracking Performance



(a) Orbits



(b) Deviations

Trust Feedback-Closed Loop Tracking Performance