Robust State Estimation and Fault Detection in CPS under Noisy Environment

C. Combastel\textsuperscript{1} A. Zolghadri\textsuperscript{1}

\textsuperscript{1}IMS Lab (UMR5218), University of Bordeaux

GT VS-CPS / MEA (GdR MACS), CNAM, Paris, 3 July 2018
Outline

1 Introduction

2 Set-membership and Probabilistic Mergers

3 Zonotopic and Gaussian Kalman Filter (ZGKF)

4 Distributed Zonotopic and Gaussian Kalman Filter (DZG-KF)
   - Problem formulation
   - Local observation and fault detection
   - Bit-level reduction and communication
   - Distributed implementation: time, synchro., schedul., data recon.

5 Numerical example

6 Conclusion
Observation, Diagnosis

Data + Knowledge → Information

Measurement + Model → State
## Observation, Diagnosis

### Paradigms to model uncertainties:
- **Stochastic** → **Kalman Filter**
- **Bounded errors** → **State bounding observer**

### Data (corrupted) + Knowledge (partial) → Information (imprecise)

<table>
<thead>
<tr>
<th>Measurement (noise) + Model (modeling errors, disturbances) → State (?)</th>
<th>Data (corrupted) + Knowledge (partial) → Information (imprecise)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Observation, Diagnosis</strong></td>
<td><strong>Numerical example</strong></td>
</tr>
<tr>
<td><strong>ZGKF (centralized)</strong></td>
<td><strong>DZG-KF (distributed)</strong></td>
</tr>
<tr>
<td><strong>Introduction</strong></td>
<td><strong>Conclusion</strong></td>
</tr>
</tbody>
</table>
Observation, Diagnosis

Paradigms to model uncertainties:

- **Stochastic** → Kalman Filter
- **Bounded errors** → State bounding observer
Paradigms to model uncertainties:
- Stochastic → Kalman Filter
- Bounded errors → State bounding observer

Both: Prediction + Correction
Prediction + Correction: with Gaussian pdf
Prediction + Correction : with Gaussian pdf
Prediction + Correction: with Gaussian pdf
Prediction + Correction : with Gaussian pdf
Prediction + Correction: with Gaussian pdf
Prediction + Correction: with interval sets
Prediction + Correction: with interval sets
Prediction + Correction: with interval sets
Prediction + Correction: with interval sets
Prediction + Correction: with interval sets
Kalman Filter: applications
Kalman Filter: applications
Distributed framework

Expected benefits [Ge et al., 2017]:

- Modularity
- Scalability
- Robustness

A set of converging technologies:

- **IoT** Internet of Things... and Everything (IoE)
- **CPS** Cyber-Physical Systems
- **MAS** Multi-Agent Systems
- **WSN** Wireless Sensor Networks
- **NCS** Network Control Systems
Distributed framework

Expected benefits [Ge et al., 2017]:

- Modularity
- Scalability
- Robustness

A set of converging technologies:

- **IoT**  Internet of Things... and Everything (IoE)
- **CPS**  Cyber-Physical Systems
- **MAS**  Multi-Agent Systems
- **WSN**  Wireless Sensor Networks
- **NCS**  Network Control Systems
Outline

1. Introduction

2. Set-membership and Probabilistic Mergers

3. Zonotopic and Gaussian Kalman Filter (ZGKF)

4. Distributed Zonotopic and Gaussian Kalman Filter (DZG-KF)
   - Problem formulation
   - Local observation and fault detection
   - Bit-level reduction and communication
   - Distributed implementation: time, synchro., schedul., data recon.

5. Numerical example

6. Conclusion
Outline

1. Introduction

2. Set-membership and Probabilistic Mergers

3. Zonotopic and Gaussian Kalman Filter (ZGKF)

4. Distributed Zonotopic and Gaussian Kalman Filter (DZG-KF)
   - Problem formulation
   - Local observation and fault detection
   - Bit-level reduction and communication
   - Distributed implementation: time, synchro., schedul., data recon.

5. Numerical example

6. Conclusion
Uncertainty paradigms

### Bounded errors
- Robustness to worst-case within specified bounds
- Known bounds, No need for pdf
- Finite supports
- Explicit domain computations

→ Disturbances (e.g. load torque)

### Stochastic
- Taking noise distribution into account
- Known pdf, Confidence level
- Possibly infinite supports
- Probabilistic evaluation of tests

→ Noise (e.g. measurements)

MERGING BOTH WORLDS
Let \((\Omega, \Sigma, \mathcal{P})\) be a probability space:
- \(\Omega\) : set of possible outcomes,
- \(\Sigma\) : collection of events,
- \(\mathcal{P}\) : probability measure.

**Probabilistic reformulation of a set-membership statement**

The set-membership statement \(x \in S\) with \(S \subset \mathbb{R}^n\) reformulates in a probabilistic framework as \(x \in \bar{R}(S)\) where \(x\) is a random variable and

\[
\bar{R}(S) = \{x : \Omega \to \mathbb{R}^n, \forall \omega \in \Omega, x(\omega) \in S\}.
\]

**Corollary: probabilistic interpretation of the set-membership “guarantee”**

\[x \in \bar{R}(S) \iff \mathcal{P}(x \in S) = 1\]

---

Definition: Set-membership and Probabilistic Merger (SPM)

A SPM is a set of random variables $x$ that can be decomposed as

$$x = d + p$$

- $d \in \overline{R}(S)$ and $S$ is a known compact set,
- $p \in \Pi$ and $\Pi$ is a set of random variables whose distribution satisfies known constraints within its (possibly infinite) support.

Example: Taking $S = \langle c, R \rangle$ and $\Pi = \overline{N}(Q)$, a SPM becomes a ZGM.
### Zonotopes

**Definition**: $p$–zonotope in $\mathbb{R}^n$

\[
\langle c, R \rangle = \{ c + Rs, \ s \in [-1, +1]^p \}
\]

- $c \in \mathbb{R}^n$: center,
- $R \in \mathbb{R}^{n \times p}$: generator segments ($p$ col.)

**Properties**:

- **Minkowski sum**: $\langle c_1, R_1 \rangle \oplus \langle c_2, R_2 \rangle = \langle c_1 + c_2, [R_1, R_2] \rangle$
- **Linear image**: $L \circ \langle c, R \rangle = \langle Lc, LR \rangle$
- **Bounding box**: $\langle c, R \rangle \subset \langle c, b(R) \rangle$, $b(R) = \text{diag}(|R|1)$
Zonotopes

**Definition**: $p$–zonotope in $\mathbb{R}^n$

\[
\langle c, R \rangle = \{ c + Rs, \ s \in [-1, +1]^p \}
\]

- $c \in \mathbb{R}^n$: center,
- $R \in \mathbb{R}^{n \times p}$: generator segments ($p$ col.)

**Properties**:

- **Minkowski sum**: $\langle c_1, R_1 \rangle \oplus \langle c_2, R_2 \rangle = \langle c_1 + c_2, [R_1, R_2] \rangle$
- **Linear image**: $L \odot \langle c, R \rangle = \langle Lc, LR \rangle$
- **Bounding box**: $\langle c, R \rangle \subset \langle c, b(R) \rangle$, $b(R) = \text{diag}(|R| \mathbf{1})$
Zonotopes: Reduction

Reduction \((q < p)\) such that \(\langle R \rangle \subset \langle \downarrow_q R \rangle\):

\[
R = [r_1, \ldots, r_j, \ldots, r_p], \quad \|r_j\|_2 \geq \|r_{j+1}\|_2;
\]

\[
R = [R_>, R_<], \quad R_> = [r_1, \ldots, r_{q-n}],
\]

\[
R_< = [r_{q-n+1}, \ldots, r_p],
\]

\[
\downarrow_q R = [R_>, b(R_<)]
\]
Zonotopes: Reduction

Reduction \((q < p)\) such that \(\langle R \rangle \subset \langle \downarrow_q R \rangle:\)

\[
R = [r_1, \ldots, r_j, \ldots, r_p], \quad \|r_j\|_2 \geq \|r_{j+1}\|_2,
\]

\[
R = [R_>, R_<], \quad R_> = [r_1, \ldots, r_{q-n}], \quad R_< = [r_{q-n+1}, \ldots, r_p],
\]

\[
\downarrow_q R = [R_>, b(R_<)],
\]
Merging set-membership and stochastic paradigms*  

Gaussian random variables with a **covariance upper bounded**† by \( Q \) (spd)

\[
\mathcal{N}(Q) = \{ x : \Omega \rightarrow \mathbb{R}^n, \exists P > 0, x \sim \mathcal{N}(0, P) \land P \preceq Q \}
\]

† Here, the covariance is *unknown* and upper bounded through a *matrix inequality*.

**Corollary 1**

\[x \in \mathcal{N}(Q) \Rightarrow \text{Cov}(x) = E[xx^T] \preceq Q\]

**Corollary 2:** Cov. matrix inequalities vs inclusion of confidence ellipsoids

\[0 \prec P \preceq Q \Rightarrow \forall c, \forall \alpha \in [0, 1], (c, P)_\alpha \subset (c, Q)_\alpha\]
Merging set-membership and stochastic paradigms*

Gaussian random variables with a covariance upper bounded\(^\dagger\) by \(Q\) (spd)

\[
\tilde{\mathcal{N}}(Q) = \{x : \Omega \to \mathbb{R}^n, \exists P > 0, x \sim \mathcal{N}(0, P) \land P \preceq Q\}
\]

\(^\dagger\) Here, the covariance is unknown and upper bounded through a matrix inequality.

Corollary 1

\[x \in \tilde{\mathcal{N}}(Q) \Rightarrow \text{Cov}(x) = \mathbb{E}[xx^T] \preceq Q\]

Corollary 2: Cov. matrix inequalities vs inclusion of confidence ellipsoids

\[0 < P \preceq Q \Rightarrow \forall c, \forall \alpha \in [0, 1], (c, P)_\alpha \subset (c, Q)_\alpha\]
Zonotopic and Gaussian Mergers (ZGM)*

Let $\mathcal{Z}(R) \triangleq \mathcal{R}(S)$ with $S = \langle 0, R \rangle$,

**Definition: Zonotopic and Gaussian Merger (ZGM)**

$$\mathcal{M}(c, R, Q) = \{ x = c + z + g, \ z \in \mathcal{Z}(R) \land g \in \mathcal{N}(Q) \}$$

- $\mathcal{Z}(R)$ is the set of random variables whose support is included in the centered zonotope $\langle 0, R \rangle$, whatever the distribution within the support is.
- $\mathcal{N}(Q)$ is the set of centered Gaussian variables with unknown but upper bounded covariance, and $Q = \text{known upper bound (matrix inequality)}$.

**Property: “Guaranteed” probability bounds for confidence sets**

$$x \in \mathcal{M}(c, R, Q) \Rightarrow \mathcal{P}(x \in \langle c, R \rangle \oplus (0, Q)_{\alpha}) \geq 1 - \alpha$$

→ Merging set-membership and stochastic paradigms in a context of imprecise probabilities where “guaranteed” inequalities about probabilities can still be derived.
Zonotopic and Gaussian Mergers (ZGM)*

**ZGM properties**

Let \( x_i \in \tilde{M}(c_i, R_i, Q_i), i \in \{\emptyset, 1, 2\} \), be such that the random variables \( x_1 \) and \( x_2 \) have statistically independent Gaussian components. Then:

\[
\begin{align*}
    x_1 + x_2 &\in \tilde{M}(c_1 + c_2, [R_1, R_2], Q_1 + Q_2), \\
    Lx &\in \tilde{M}(Lc, LR, LQL^T), \\
    P(x \in (c \pm r_\alpha)) &\geq 1 - \alpha,
\end{align*}
\]

where \( r_\alpha = |R|1 + \text{diag}\frac{1}{2}(Q\chi_n^2(1 - \alpha)) \in \mathbb{R}^n \).

- \( \chi_n^2(1 - \alpha) \in \mathbb{R} \): value taken for the probability \( 1 - \alpha \) by the quantile function of the chi-squared distribution with \( n \) degrees of freedom.
- \( \alpha \in [0, 1] \) interprets as a maximal probability of type I error (false alarm rate).
Outline

1. Introduction
2. Set-membership and Probabilistic Mergers
3. Zonotopic and Gaussian Kalman Filter (ZGKF)
4. Distributed Zonotopic and Gaussian Kalman Filter (DZG-KF)
   - Problem formulation
   - Local observation and fault detection
   - Bit-level reduction and communication
   - Distributed implementation: time, synchro., schedul., data recon.
5. Numerical example
6. Conclusion
Problem: state estimation and fault detection

LTV fault-free model

\[ x_{k+1} = A_k x_k + B_k u_k + v_k \]
\[ y_k = C_k x_k + D_k u_k + w_k \]

\[ x_0 \in \tilde{M}(c_0, R_0, Q_0) \]
\[ v_k \in \tilde{M}(0, E_z,k, E_g,k E_{g,k}^T) \]
\[ w_k \in \tilde{M}(0, F_z,k, F_g,k F_{g,k}^T) \]

Initial state uncertainty:

\[ x_0 = c_0 + z_0 + g_0 \]
\[ z_0 \in \tilde{Z}(R_0), \quad g_0 \in \tilde{N}(Q_0) \]

Gaussian noise terms are all assumed to be mutually independent.
Problem: state estimation and fault detection

LTV fault-free model (simplified notations: no $k$ but still LTV)

\[
\begin{align*}
\mathbf{x}_+ &= A\mathbf{x} + Bu + \mathbf{v} \\
y &= C\mathbf{x} + Du + \mathbf{w} \\
\mathbf{x}_0 &\in \tilde{\mathcal{M}}(c_0, R_0, Q_0) \\
\mathbf{v} &\in \tilde{\mathcal{M}}(0, E_z, E_g E_g^T) \\
\mathbf{w} &\in \tilde{\mathcal{M}}(0, F_z, F_g F_g^T)
\end{align*}
\]

Initial state uncertainty:

\[
\begin{align*}
\mathbf{x}_0 &= c_0 + \mathbf{z}_0 + \mathbf{g}_0 \\
\mathbf{z}_0 &\in \tilde{\mathcal{Z}}(R_0), \quad \mathbf{g}_0 \in \tilde{\mathcal{N}}(Q_0)
\end{align*}
\]

Gaussian noise terms are all assumed to be mutually independent.
After introducing a time-varying observer gain $G$ (Luenberger-like), and starting from $k = 0$, the iterations $\forall k \geq 0$ rely on:

$$\mathbf{x} \in \bar{M}(c, R, Q) \Downarrow$$

$$\mathbf{x}_+ \in \bar{M}(c_+, R_+, Q_+)$$

where:

$$c_+ = (A - GC)c + (B - GD)u + Gy, \quad (1)$$

$$R_+ = [(A - GC)\bar{R}, E_z, -GF_z], \quad \bar{R} = \downarrow_q R, \quad (2)$$

$$Q_+ = (A - GC)Q(A - GC)^T + E_gE_g^T + GFG_g^TGT. \quad (3)$$

$(1), (2), (3)$ : center, zonotopic, covariance updates.
Multi-objective optimality criterion

\[ \mathbf{x}_+ = c_+ + \mathbf{z}_+ + \mathbf{g}_+, \quad \forall k \]

\[ \mathbf{z}_+ \in \tilde{Z}(R_+(G)), \quad \mathbf{g}_+ \in \tilde{N}(Q_+(G)) \]

Question: Time-varying observer gain \( G \) giving the “best” state estimate?

### Zonotopic uncertainty

Minimize trace of covariation:

\[ J_z = \text{tr}(\text{cov}(\mathbf{z}_+)) = \text{tr}(P_+), \]

\[ P_+ = R_+R_+^T \]

### Gaussian uncertainty

Minimize trace of covariance:

\[ J_g = \text{tr}(\text{Cov}(\mathbf{g}_+)) = \text{tr}(Q_+) \]

(idem KF)

### Joint minimization of covariation and covariance:

\[ J = (1 - \eta)J_z + \eta J_g \]

\[ \eta \in [0, 1] : \text{weighting constant, e.g. } \eta = \frac{1}{2} \]
Theorem: Optimal observer gain

The time-varying optimal observer gain $G^* = \arg\min_G J$ jointly minimizing Zonotopic and Gaussian uncertainties about the states is:

$$G^* = AK^*, \quad K^* = LS^{-1},$$

$$L = Q_x C^T, \quad S = CQ_x C^T + Q_w.$$ 

$$Q_x = (1 - \eta) \bar{R}\bar{R}^T + \eta Q$$

$$Q_w = (1 - \eta) F_z F_z^T + \eta F_g F_g^T$$

A remarkable feature of ZGKF: the richer uncertainty modeling granted by zonotopes leaves $K^*$ completely analog to the usual Kalman gain!
Recall: \( J = (1 - \eta) J_z + \eta J_g \)

- If \( \eta = 1 \) with no bounded disturbances \( (E_z = 0, F_z = 0, R_0 = 0) \), then ZGKF reduces to the usual discrete-time Kalman Filter (KF) in one-step ahead predictor form (easy to embed in control loops),

- If \( \eta = 0 \) with no Gaussian noise \( (E_g = 0, F_g = 0, Q_0 = 0) \) and no reduction \( (\bar{R} = R) \), then ZGKF again reduces to KF, at the price of losing the zonotopic shape matrix \( R \). Keeping \( R \) with reduction leads to ZKF (Automatica, 2015)\(^1\).

Outline

1. Introduction
2. Set-membership and Probabilistic Mergers
3. Zonotopic and Gaussian Kalman Filter (ZGKF)
4. Distributed Zonotopic and Gaussian Kalman Filter (DZG-KF)
   - Problem formulation
   - Local observation and fault detection
   - Bit-level reduction and communication
   - Distributed implementation: time, synchro., schedul., data recon.
5. Numerical example
6. Conclusion
DZG-KF requirements:

- Distributed state observation over interconnected agents
- Interactions through both physical and communication links (CPS)
- Robust fault detection up to a given max. false alarm rate ...
- ... under both set-membership and random uncertainties
- Compatible with distributed control e.g. local state feedbacks
- Taking networking constraints into account:
  - Time management
  - Parallelism/concurrency
  - Potential packet losses
  - Sustainability (reduction of required com. channel capacities)

C. Combastel, A. Zolghadri, FDI in Cyber Physical Systems: A Distributed Zonotopic and Gaussian Kalman Filter with Bit-level Reduction, IFAC SAFEPROCESS 2018.
Cyber-physical agent $A_i$, $i = 1, \ldots, n_a$

**LTV fault-free model of $A_i$**

$$
\begin{align*}
    x_{i,k+1} &= A_{ii,k} x_{i,k} + \sum_{j \in N_i} A_{ij,k} x_{j,k} + B_{i,k} u_{i,k} + v_{i,k}, \\
    y_{i,k} &= C_{i,k} x_{i,k} + D_{i,k} u_{i,k} + w_{i,k},
\end{align*}
$$

where $x_{i,0} \in \tilde{M}(c_{i,0}, R_{i,0}, Q_{i,0})$, $v_{i,k} \in \tilde{M}(0, E_{z,i,k}, E_{g,i,k} E_{g,i,k}^T)$, $w_{i,k} \in \tilde{M}(0, F_{z,i,k}, F_{g,i,k} F_{g,i,k}^T)$.

where Gaussian noise terms are all assumed to be mutually independent.
Cyber-physical agent $A_i$, $i = 1, \ldots, n_a$

**LTV fault-free model of $A_i$** (simplified notations: no $k$ but still LTV)

\[
\begin{align*}
    x_{i,+} &= A_{ii}x_i + \sum_{j \in N_i} A_{ij}x_j + B_i u_i + v_i, \\
    y_i &= C_i x_i + D_i u_i + w_i,
\end{align*}
\]

\[
\begin{align*}
    x_{i,0} &\in \tilde{M}(c_i,0, R_i,0, Q_i,0), \\
    v_i &\in \tilde{M}(0, E_{z,i}, E_{g,i}E_{g,i}^T), \\
    w_i &\in \tilde{M}(0, F_{z,i}, F_{g,i}F_{g,i}^T).
\end{align*}
\]

where Gaussian noise terms are all assumed to be mutually independent.
DZG-KF: Code embedded on each autonomous agent

1: procedure Agent(i)
2: \((X_i,0,\alpha_i,q_i,w_i,X^\theta_i) = \texttt{INITIALIZE}(i)\)
3: \(N_i = \texttt{GetNeighbors}(i)\)
4: \((X^\theta_j)_{j \in N_i} = \texttt{NEGOTIATE\_RECONCILIATION}(N_i, X^\theta_i)\) ▷ \(X^\theta_j\): def. data (agent \(j\))
5: \(\texttt{SYNCHRONIZE\_CLOCK}(i, N_i)\)
6: \(\texttt{IDLE\_UNTIL}(t == 0)\) ▷ start event
7: \(k = 0, X_i = X_i,0\) ▷ initial sample and state
8: loop ▷ \(i\)th agent’s main loop
9: \(y_i = \texttt{LOCAL\_DATA\_ACQUIS}\)
10: \(\texttt{LOCAL\_ACTUATION}(u_i)\)
11: \((c_i, R_i, Q_i) \leftarrow X_i\)
12: \(\bar{R}_i = \downarrow q_i R_i\)
13: \(\bar{X}_i = (c_i, \bar{R}_i, Q_i)\)
14: \(T_i = \texttt{LOCAL\_DETECT}(y_i, \bar{X}_i, \alpha_i)\)
15: \((\bar{X}_j)_{j \in N_i} = \texttt{BIT\_RED\_COM}(\bar{X}_i)\)
16: \(X_i,+ = \texttt{LOCAL\_ZGKF}(\bar{X}_i, (\bar{X}_j)_{j \in N_i}, y_i, u_i, k)\) ▷ update and prediction
17: \(u_i,+ = \texttt{LOCAL\_CONTROL}(X_i,+ )\)
18: \(\texttt{IDLE\_UNTIL}(t == (k + 1)h)\)
19: \(k = k + 1, X_i = X_i,+ , u_i = u_i,+\)
20: end loop
21: end procedure
DZG-KF : Code embedded on each autonomous agent

1: procedure Agent(i)
2:  (X_i,0, α_i, q_i, w_i, X_0^i) = INITIALIZE(i)
3:  N_i = GetNeighbors(i)
4:  (X^0_j)_{j∈N_i} = NEGOTIATERECONCILIATION(N_i, X^0_i) \triangleright X^0_j: \text{def. data (agent } j) \triangleright t : \text{physical time}
5:  SYNCHRONIZEClock(i, N_i)
6:  IDLEUntil(t == 0)
7:  k = 0, X_i = X_i,0 \triangleright \text{initial sample and state}
8:  loop
9:     y_i = LOCALDataAcquis
10:    LOCALActuation(u_i)
11:     (c_i, R_i, Q_i) \leftarrow X_i \triangleright \text{state: } X_i(k|k-1)
12:     \tilde{R}_i = ↓q_i R_i \triangleright \text{reduce } R_i(k|k-1)
13:     \tilde{X}_i = (c_i, \tilde{R}_i, Q_i)
14:     T_i = LOCALDetect(y_i, \tilde{X}_i, α_i) \triangleright \text{ZGM used to...}
15:     (\tilde{X}_j)_{j∈N_i} = \text{BitRedCom}(\tilde{X}_i) \triangleright \text{detect and}
16:     \triangleright \text{communicate}
17:     X_i, + = LOCALZGKF(\tilde{X}_i, (\tilde{X}_j)_{j∈N_i}, y_i, u_i, k) \triangleright \text{update and prediction}
18:     \triangleright \text{give } X_i, + = X_i(k+1|k)
19:     \triangleright \text{next control}
20:     \triangleright \text{next sample}
21:     \triangleright \text{next iteration}
22:     u_i, + = LOCALControl(X_i, +)
23:     IDLEUntil(t == (k + 1)h)
24:     k = k + 1, X_i = X_i, +, u_i = u_i, +
25: end loop
26: end procedure
DZG-KF: Code embedded on each autonomous agent

1: procedure Agent(i) ▷ ith agent’s code
2: \((X_i, 0, \alpha_i, q_i, w_i, X_i^0) = \text{INITIALIZE}(i)\)
3: \(N_i = \text{GETNEIGHBORS}(i)\)
4: \((X_j^0)_{j \in N_i} = \text{NEGOTIATERECONCILIATION}(N_i, X_i^0)\) ▷ \(X_j^0\): def. data (agent \(j\))
5: \(\text{SYNCHRONIZECLOCK}(i, N_i)\)
6: \(\text{IDLEUNTIL}(t == 0)\) ▷ \(t\): physical time ▷ start event
7: \(k = 0, \quad X_i = X_i, 0\) ▷ initial sample and state ▷ ith agent’s main loop
8: loop
9: \(y_i = \text{LOCALDATAACQUIS}\) ▷ measure: \(y_i(k)\)
10: \(\text{LOCALACTUATION}(u_i)\) ▷ control: \(u_i(k)\)
11: \((c_i, R_i, Q_i) \leftarrow X_i\) ▷ state: \(X_i(k|k-1)\)
12: \(\tilde{R}_i = \downarrow q_i R_i\) ▷ reduce \(R_i(k|k-1)\)
13: \(\tilde{X}_i = (c_i, \tilde{R}_i, Q_i)\) ▷ ZGM used to… ▷ detect and communicate
14: \(T_i = \text{LOCALDETECT}(y_i, \tilde{X}_i, \alpha_i)\)
15: \((\tilde{X}_j)_{j \in N_i} = \text{BITREDCOM}(\tilde{X}_i)\)
16: \(X_i, + = \text{LOCALZGKF}(\tilde{X}_i, (\tilde{X}_j)_{j \in N_i}, y_i, u_i, k)\) ▷ update and prediction ▷ give \(X_i, + = X_i(k+1|k)\)
17: ▷ next control ▷ next sample ▷ next iteration
18: \(u_i, + = \text{LOCALCONTROL}(X_i, +)\)
19: \(\text{IDLEUNTIL}(t == (k + 1)h)\)
20: \(k = k + 1, \quad X_i = X_i, +, \quad u_i = u_i, +\)
21: end loop
22: end procedure
DZG-KF : Code embedded on each autonomous agent

1: procedure Agent(i) \(\triangleright\) \(i\)th agent’s code
2: \((X_i, 0, \alpha_i, q_i, w_i, X_i^0) = \text{initialize}(i)\)
3: \(N_i = \text{GetNeighbors}(i)\)
4: \((X_j^0)_{j \in N_i} = \text{NegotiateReconciliation}(N_i, X_i^0)\) \(\triangleright\) \(X_j^0\) : def. data (agent \(j\))
5: \(\text{SynchronizeClock}(i, N_i)\)
6: \(\text{IdleUntil}(t == 0)\)
7: \(k = 0, \ X_i = X_i, 0\)
8: loop \(\triangleright\) \(i\)th agent’s main loop
9: \(y_i = \text{LocalDataAcquis}\)
10: \(\text{LocalActuation}(u_i)\)
11: \((c_i, R_i, Q_i) \leftarrow X_i\)
12: \(\bar{R}_i = \downarrow q_i R_i\)
13: \(\bar{X}_i = (c_i, \bar{R}_i, Q_i)\)
14: \(T_i = \text{LocalDetect}(y_i, \bar{X}_i, \alpha_i)\)
15: \((\bar{X}_j)_{j \in N_i} = \text{BitRedCom}(\bar{X}_i)\)
16: \(X_{i,+} = \text{LocalZGKF}(\bar{X}_i, (\bar{X}_j)_{j \in N_i}, y_i, u_i, k)\) \(\triangleright\) update and prediction
17: \(u_{i,+} = \text{LocalControl}(X_{i,+})\) \(\triangleright\) next control
18: \(\text{IdleUntil}(t == (k + 1)h)\)
19: \(k = k + 1, \ X_i = X_{i,+}, \ u_i = u_{i,+}\)
20: end loop
21: end procedure
Problem formulation

DZG-KF: Code embedded on each autonomous agent

1: procedure Agent(i)  ▶ ith agent’s code
2: \((X_{i,0}, \alpha_i, q_i, w_i, X^\theta_i) = \text{INITIALIZE}(i)\)
3: \(N_i = \text{GETNEIGHBORS}(i)\)
4: \((X^\theta_j)_{j \in N_i} = \text{NEGOTIATERECONCILIATION}(N_i, X^\theta_i)\) ▶ \(X^\theta_j\): def. data (agent \(j\))
5: \(\text{SYNCHRONIZECLOCK}(i, N_i)\)  ▶ \(t\): physical time
6: \(\text{IDLEUNTIL}(t == 0)\) ▶ start event
7: \(k = 0, \ X_i = X_{i,0}\) ▶ initial sample and state
8: loop ▶ ith agent’s main loop
9: \(y_i = \text{LOCALDATAACQUIS}\) ▶ measure: \(y_i(k)\)
10: \(\text{LOCALACTUATION}(u_i)\) ▶ control: \(u_i(k)\)
11: \((c_i, R_i, Q_i) \leftarrow X_i\) ▶ state: \(X_i(k|k-1)\)
12: \(\bar{R}_i = \downarrow q_i R_i\) ▶ reduce \(R_i(k|k-1)\)
13: \(\bar{X}_i = (c_i, \bar{R}_i, Q_i)\) ▶ \(\text{ZGM used to. . .}\)
14: \(T_i = \text{LOCALDETECT}(y_i, \bar{X}_i, \alpha_i)\) ▶ detect and
15: \((\bar{X}_j)_{j \in N_i} = \text{BITREDCOM}(\bar{X}_i)\) ▶ communicate
16: \(X_{i,+} = \text{LOCALZGKF}(\bar{X}_i, (\bar{X}_j)_{j \in N_i}, y_i, u_i, k)\) ▶ update and prediction
17: \(u_{i,+} = \text{LOCALCONTROL}(X_{i,+})\) ▶ give \(X_{i,+} = X_i(k+1|k)\)
18: \(\text{IDLEUNTIL}(t == (k + 1)h)\) ▶ next control
19: \(k = k + 1, \ X_i = X_{i,+}, \ u_i = u_{i,+}\) ▶ next sample
20: end loop ▶ next iteration
21: end procedure
Local Observation (ZGKF)

1: function $X_{i,+} = \text{LOCAL ZGKF}(\bar{X}_i, (\bar{X}_j)_{j \in N_i}, y_i, u_i, k)$

2: $(c_i, \bar{R}_i, Q_i) \leftarrow \bar{X}_i$ \hspace{1cm} $\triangleright$ state prediction: $\bar{X}_i(k|k-1)$

3: $\bar{P}_i = \bar{R}_i \bar{R}_i^T$ \hspace{1cm} $\triangleright$ covariation: $\bar{P}_i(k|k-1)$

4: $Q_{vz,i} = E_{z,i} E_{z,i}^T, \quad Q_{vg,i} = E_{g,i} E_{g,i}^T, \quad Q_{wz,i} = F_{z,i} F_{z,i}^T, \quad Q_{wg,i} = F_{g,i} F_{g,i}^T$

5: $\bar{Q}_x,i = (1 - \eta) \bar{P}_i + \eta Q_i, \quad \bar{Q}_{w,i} = (1 - \eta) Q_{wz,i} + \eta Q_{wg,i}$

6: $S_i = C_i \bar{Q}_x,i C_i^T + Q_{w,i}$

7: $K_i = Q_x,i C_i^T S_i^{-1}$ \hspace{1cm} $\triangleright$ Kalman gain

8: $G_i = A_{ii} K_i$ \hspace{1cm} $\triangleright$ Observer gain

9: $\bar{A}_i = A_{ii} - G_i C_i, \quad \bar{B}_i = B_i - G_i D_i$

10: $r_{i,+} = \bar{A}_i c_i + \bar{B}_i u_i + G_i y_i$

11: $R_{i,+} = [\bar{A}_i \bar{R}_i, E_{z,i}, -G_i F_{z,i}]$

12: $Q_{i,+} = \bar{A}_i Q_i \bar{A}_i^T + Q_{vz,i} + G_i Q_{wg,i} G_i^T$ \hspace{1cm} $\triangleright$ Center

13: for $j \in N_i$ do

14: $(c_j, \bar{R}_j, Q_j) \leftarrow \bar{X}_j$

15: $c_{i,+} = c_{i,+} + A_{ij} c_j$ \hspace{1cm} $\triangleright$ Zono.

16: $R_{i,+} = [R_{i,+}, A_{ij} \bar{R}_j]$ \hspace{1cm} $\triangleright$ Gauss.

17: $Q_{i,+} = Q_{i,+} + A_{ij} Q_j A_{ij}^T$ \hspace{1cm} $\triangleright$ for each neighbor

18: end for

19: $X_{i,+} = (c_{i,+}, R_{i,+}, Q_{i,+})$

20: return $X_{i,+}$

21: end function
Local Detection Test

1. **function** \( T_i = \text{LOCALDETECT}(y_i, \bar{X}_i, \alpha_i) \)
2. \((c_i, \bar{R}_i, Q_i) \leftarrow \bar{X}_i\)
3. \(\varepsilon_i = y_i - C_i c_i - D_i u_i\)  \(\triangleright\) Innovation/residual
4. \(\tilde{R}_i = [C_i \bar{R}_i, F_{z,i}]\)  \(\triangleright\) Zono.
5. \(\tilde{Q}_i = C_i Q_i C_i^T + Q_{w, i}\)  \(\triangleright\) Gauss.
6. \(r_{\alpha_i} = |\tilde{R}_i| \mathbf{1} + \text{diag}^{\frac{1}{2}} (\tilde{Q}_i \chi_{n_{y_i}}^2 (1 - \alpha_i))\)  \(\triangleright\) Adaptive threshold
7. \(T_i = (\varepsilon_i \notin 0 \pm r_{\alpha_i})\)  \(\triangleright\) Test s.t. \(ok \Rightarrow P(T_i) \leq \alpha_i\)
8. **end function**
Definition [IFAC SAFEPROCESS’2018] : Bit-level reduction

The bit-level reduction of a set $S$ is an operator denoted $\downarrow_w$ taking as input some data used to define $S$ and returning data reducing the binary footprint (i.e. quantity of information in bits) of the input to some fixed value parameterized by $w$, while accurately approximating $S$ up to some quantified error bound.

- Example: Quantization of a zonotope shape matrix + error bound:

$$R \in \mathbb{F}^{n \times p} \text{ (floating point)} \approx (R^d)2^{e_R} \text{ (fixed point)}, \quad R^d \in \mathbb{I}_w^{n \times p},$$

$$\langle R \rangle \subset \langle (R^d)2^{e_R} \rangle \oplus (0 \pm (b^d)2^{e_b})$$

- Example: $\sigma(\mathbb{F}) = 64$, $w = 16 \Rightarrow \rho \lesssim 4$ (e.g. 3.86 for $n = 5$ and $p = 40$)
Outline

1. Introduction
2. Set-membership and Probabilistic Mergers
3. Zonotopic and Gaussian Kalman Filter (ZGKF)
4. Distributed Zonotopic and Gaussian Kalman Filter (DZG-KF)
   - Problem formulation
   - Local observation and fault detection
   - Bit-level reduction and communication
   - Distributed implementation: time, synchro., schedul., data recon.
5. Numerical example
6. Conclusion
Physical topology of the spring system

Figure: Physical topology of the spring system.
Communication network topology

Figure: Communication network topology.

Topology:
- Not fully interconnected
- Not linear
- No agent has a global view from its neighbors
Spring system data

Agents Modeling:

- Agents states: \( x_i = [p_i; q_i] \) (pos., speed). \( n_{xi} = 2 \) or \( 4 \) for \( i = 5, 6 \)
- Param. (Sl): \( m_i = 1, l_{ij} = 1, \kappa_{ij} = 1 \) or \( 1.5 \) for \( ij = 15, 36, \nu_{ij} = 0.1 \)
- Linearization around the equilibrium points
- Accelerations: all subject to \( \pm 0.001 \) bounded error, no Gaussian noise
- Sensors: None for \( A_1, A_4 \) (trivially non observable). Other agents: position sensor with Gaussian noise (\( 3\sigma \leq 1mm \)).
- Sampled time scale: \( h = 0.05, k_{max} = 3000 \)
- Initial state uncertainties: bounded by \( \pm 0.12 \) (pos.) or \( \pm 0.001 \) (speeds)

“True” system simulation:

- Initial state: pos. deviation from eq. of 0.1 for \( A_1 \), 0 for \( A_i, i \neq 1 \).
- Actuation: horiz. step force \( u_1 = 0.03 \) applied to \( A_1 \) at \( k = 1500 \)
- Actuator faults: \( f_i = [0; 0.005] \) in the state eq. of \( A_i \) at \( k > 1000 \)

DZG-KF: \( w_i = 16 \) bits, max. f.a.r. \( \alpha_i = 0.0027 \) i.e. \( \pm 3\sigma \) in scalar cases
State estimation (1/2)

**Figure:** Estimation of the $x$-axis position of node 6 by $\mathcal{A}_6$.

**Figure:** Estimation of the $y$-axis position of node 6 by $\mathcal{A}_6$. 
State estimation (2/2)

Figure: Estimation of the $(x)$-axis position of node 1 by $A_1$.

Figure: Estimation of the $x$-axis speed of node 5 by $A_5$. 
Fault detection

**Figure:** Fault $f_2$ in node 2: agents local detection tests $T_i$.

**Figure:** Fault $f_1$ in node 1: agents local detection tests $T_i$. 
Outline

1. Introduction

2. Set-membership and Probabilistic Mergers

3. Zonotopic and Gaussian Kalman Filter (ZGKF)

4. Distributed Zonotopic and Gaussian Kalman Filter (DZG-KF)
   - Problem formulation
   - Local observation and fault detection
   - Bit-level reduction and communication
   - Distributed implementation: time, synchro., schedul., data recon.

5. Numerical example

6. Conclusion
Conclusion about DZG-KF

- Distributed state observation
- Interactions through both physical and communication links (CPS)
- Robust fault detection up to a given max. false alarm rate . . .
- . . . under both set-membership and random uncertainties
- Compatible with distributed control
- Autonomous agents networking:
  - Time management  \rightarrow  Clocks synchronization
  - Parallelism/concurrency  \rightarrow  IdleUntil, schedulability
  - Potential packet losses  \rightarrow  Data reconciliation strategy
  - Sustainability  \rightarrow  Bit-level reduction

Selected publications (DZG-KF - EZGKF - ZGKF - ZKF):


Thank you!

**Selected publications (DZG-KF - EZGKF - ZGKF - ZKF):**


