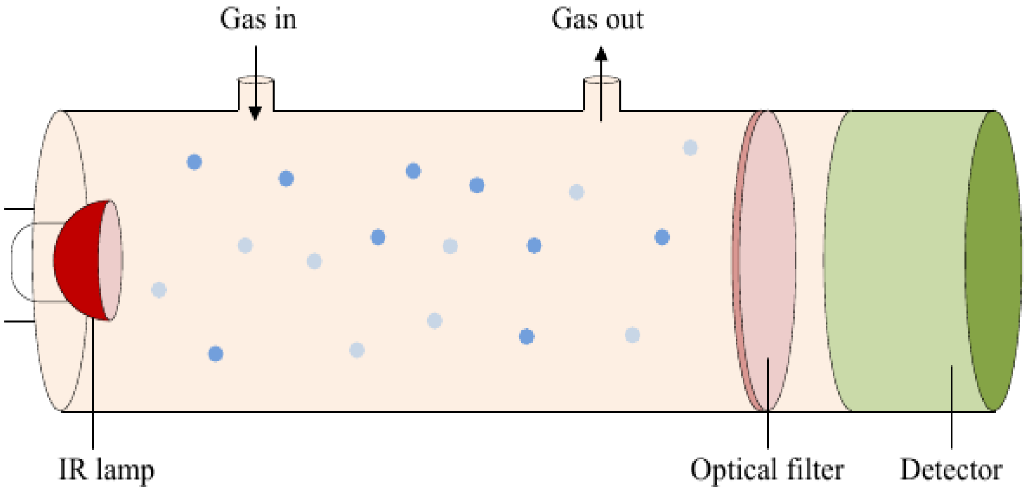
# Belief Function Fusion based Self-calibration for Non-dispersive Infrared Gas Sensor

Yang You and Tobias Oechtering
Division of Information Science and Engineering, EECS
KTH, Royal Institute of Technology, Sweden

# NDIR $CO_2$ Sensor

- Sensor Mechanism:
  - Different $CO_2$ molecules → Absorption of light with different wavelengths
  - Attenuation of light intensity → Absorption of the measured $CO_2$ → $CO_2$ concentration
- Measurement process:
  - $Abs = f(zero, IR, T)$
  - $CO_2 = L(R - Abs)$
  - $Abs$: Absorption
  - IR: The amount of received IR light
  - Zero: Zero coefficient
  - R: Reference level



# Drift Analysis of NDIR Sensor

- Zero coefficient:
  - Calibration parameter used for adjusting the sensor baseline offset
- Why there is drift?
  - IR signal varies according to time, temperature and/or other factors
  - But the same zero coefficient is being used for calculating absorption.
  - Lead to a measurement error
- Sensor calibration requires adjusting the zero coefficient
  - Build a stochastic model of the true zero coefficient
  - Estimate the true zero coefficient at any time given noisy observation

# Automatic Baseline Correction (ABC)

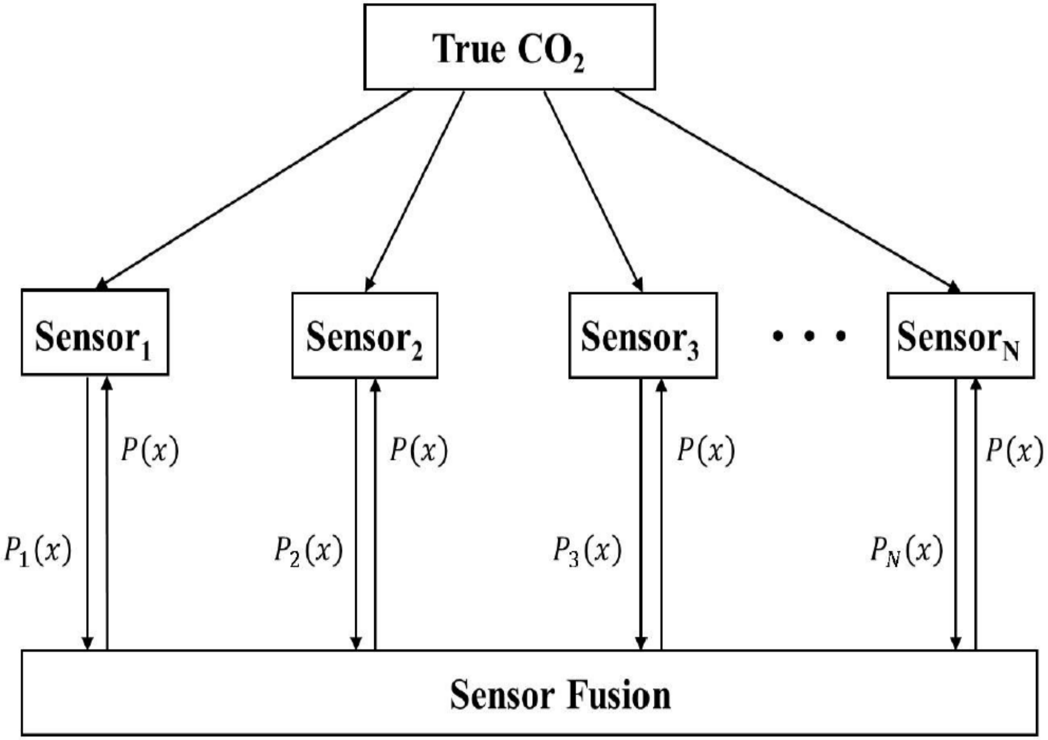
- In each calibration period:
  - The sensor is calibrated to a fixed value which is assumed to be the fresh air $CO_2$ concentration
- Fails when the sensors never get exposed to fresh air in a calibration period
  - E.g., in mega-cities
- Need to design more robust and smart self-calibration algorithms

# System Model and Problem Formulation

- An NDIR sensor system：
  - $N$ sensors measure time-varying $CO_2$ level
    - Same environment: Same target for all sensors
  - Posterior distribution of the current true $CO_2$ level given historical measurements at each sensor
    - Via hidden Markov model, our previous work [1]
    - Named as belief function in this work.
- Sensor calibration:
  - Fuse the belief functions of each sensor.
  - The fused belief function can be adopted as the new belief function by all sensors

# System Model and Problem Formulation

- True $CO_2$ level: $x \in \bar{X}$
  - $\bar{X} = \{x_1, x_2, \ldots, x_M\}$
- Belief function of sensor $i$ on the true $CO_2$ level: $P_i(x)$
- Fused belief function: $P(x)$



# Belief Function Fuse via Dempster's Rule

- Assumption: Belief functions for all $N$ sensors are reliable
- Two sensor fusion case:
  - Consider $sensor_i$ and $sensor_j$

$$\left(P_i \oplus P_j\right)(x_k) = \frac{1}{1-F} P_i(x_k) P_j(x_k), \forall x_k \in \bar{X}$$

  - $F = \sum_{x_m, x_n \in \bar{X}, x_m \neq x_n} P_i(x_m) P_j(x_n)$
- $N$ sensors fusion case:

$$\begin{aligned} P(x_k) &= (P_1 \oplus P_2 \oplus \cdots \oplus P_N)(x_k) \\ &= (P_1(x_k) P_2(x_k) \dots P_N(x_k))/F', \forall x_k \in \bar{X} \end{aligned}$$

  - $F' = \sum_{x_m \in \bar{X}} P_1(x_m) P_2(x_m) \dots P_N(x_m)$
- $P(x)$ is then further used by all sensors

# Belief Function Fusion via Wasserstein Distance based Weighted Average

- Dempster's rule fails when the belief functions highly conflict with each other.
  - Need some pre-processing of the original belief functions
  - Weighted average approach
- Wasserstein distance:
  - Two random variables $Y$ and $Z$ with distribution $P_Y$ and $P_Z$

$$W_2(P_Y, P_Z) = \sqrt{\min_{P_{YZ}: \sum_z P_{YZ} = P_Y, \sum_y P_{YZ} = P_Z} \sum_{y \in \mathcal{Y}, z \in \mathcal{Z}} |y - z|^2 P_{YZ}(y, z)}.$$

  - Distance between belief functions of $sensor_i$ and $sensor_j$
    - $W_2\left(P_i(x), P_j(x)\right), \forall i, j \in \{1,2, \dots, N\}$

# Belief Function Fusion via Wasserstein Distance based Weighted Average

- Normalized distance in interval [0,1]:

$$\hat{W}_2(P_i(x), P_j(x)) = \frac{2 \times W_2(P_i(x), P_j(x))}{\sum_i \sum_j W_2(P_i(x), P_j(x))}$$

- Similarity:

$$S(P_i(x), P_j(x)) = 1 - \hat{W}_2(P_i(x), P_j(x))$$

- Support degree (importance of the belief function):

$$Supp(P_i(x)) = \sum_{j=1, j \neq i}^{N} S(P_i(x), P_j(x))$$

# Belief Function Fusion via Wasserstein Distance based Weighted Average

- Calculation of the weight:

$$\alpha_i = \frac{Supp(P_i(x))}{\sum_{i=1}^{N} Supp(P_i(x))}$$

- Weighted average of $N$ belief functions:

$$\hat{P}(x) = \sum_{i=1}^{N} \alpha_i P_i(x)$$

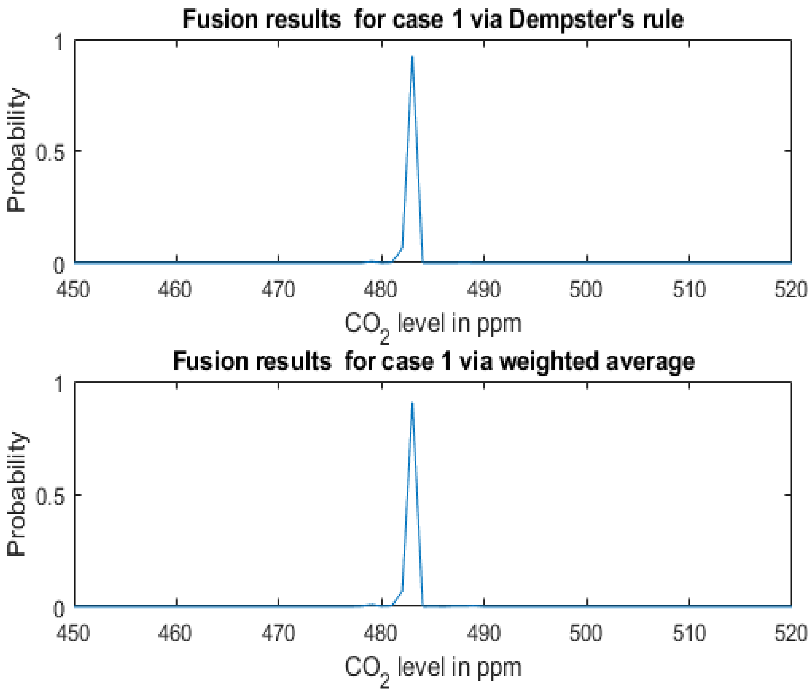
- Fused belief function:

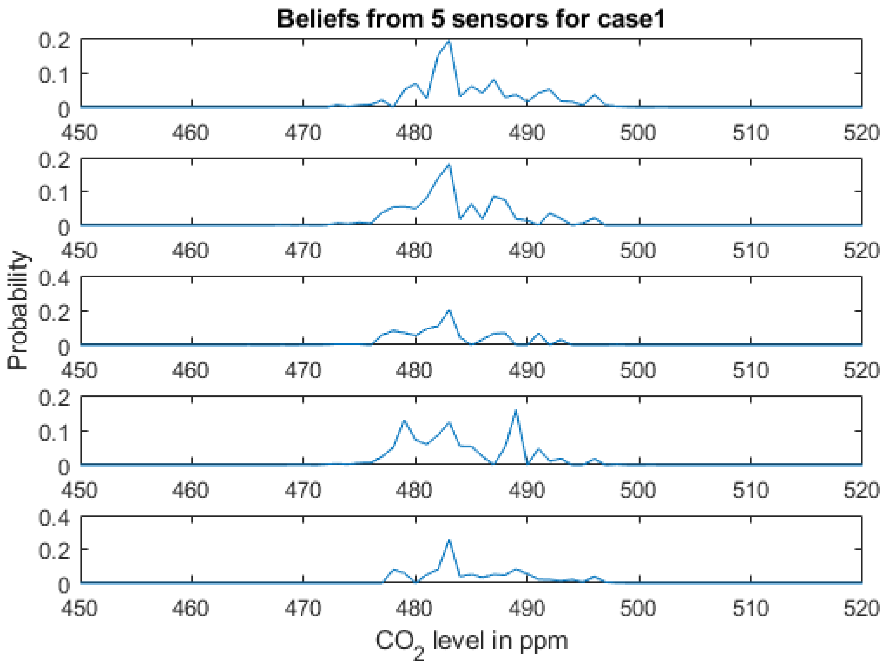
$$P(x) = (\hat{P} \oplus \hat{P} \oplus \cdots \oplus \hat{P})(x)$$

  - The operator $\oplus$ is applied for $N - 1$ times

# Numerical Results

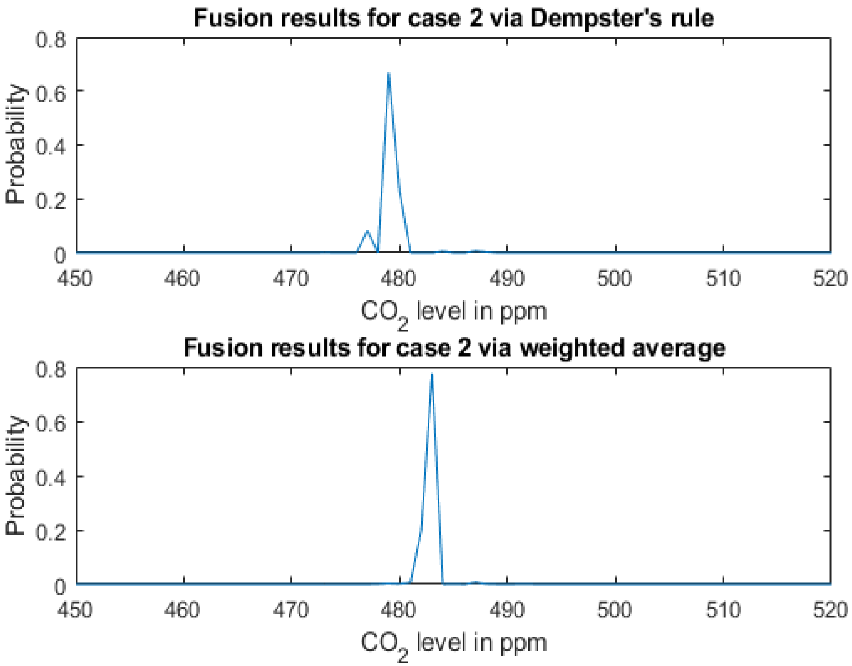
- Case 1: All belief functions are consistent, no strong conflicts.

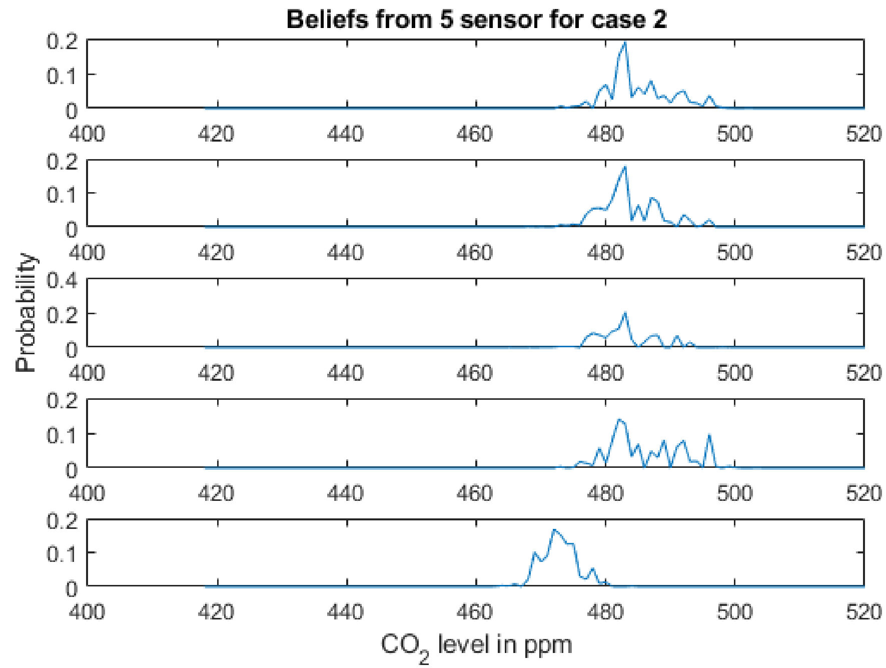


# Numerical Results

- Case 2: One sensor has strong conflict with the other sensors.



# Conclusions and Future Works

- Conclusions:
  - The general belief function fusion framework can work well in the case where no strong conflict happen
  - Weighted average approach has better performance when dealing with conflicts
- Future works:
  - Expanding our numerical experiments to more datasets and scenarios to check the robustness of our algorithms
  - Comparing the current proposed distance metric to the existing distance metrics.