Multi-Sensor Data and Information Fusion

Ning Xiong

Mälardalen University

Agenda

- What is (Multi-sensor) data fusion and why?
- Application areas of data fusion
- Two data processing architectures and related techniques for data fusion
- Fusion functionality in the JDL data fusion model

What is data fusion?

Fundamental Motivation

- The basic reason for data and information fusion is to reduce the uncertainty in our understanding and prediction of the state of certain aspect of the world.

- If the information of interest were precisely available from a single source, there would be no need to fuse with other sources.

What is multisensor data fusion?

- **Data fusion is synergistic combination of data or information from multiple sensors to get more reliable and accurate information in estimating or predicting entity states.**

Combine data

From sensors of the same kind

Form different type sensors

Estimate or predict the state of some aspect of the world

Multisensor advantages

Data fusion techniques combine data from multiple sensors to achieve more accurate and specific inferences than could be achieved by using a single sensor alone

- Identical sensors increase reliability of information
- Different sensors produce complementary information, broadens the observation for better decision

Application areas of data fusion

Defense applications

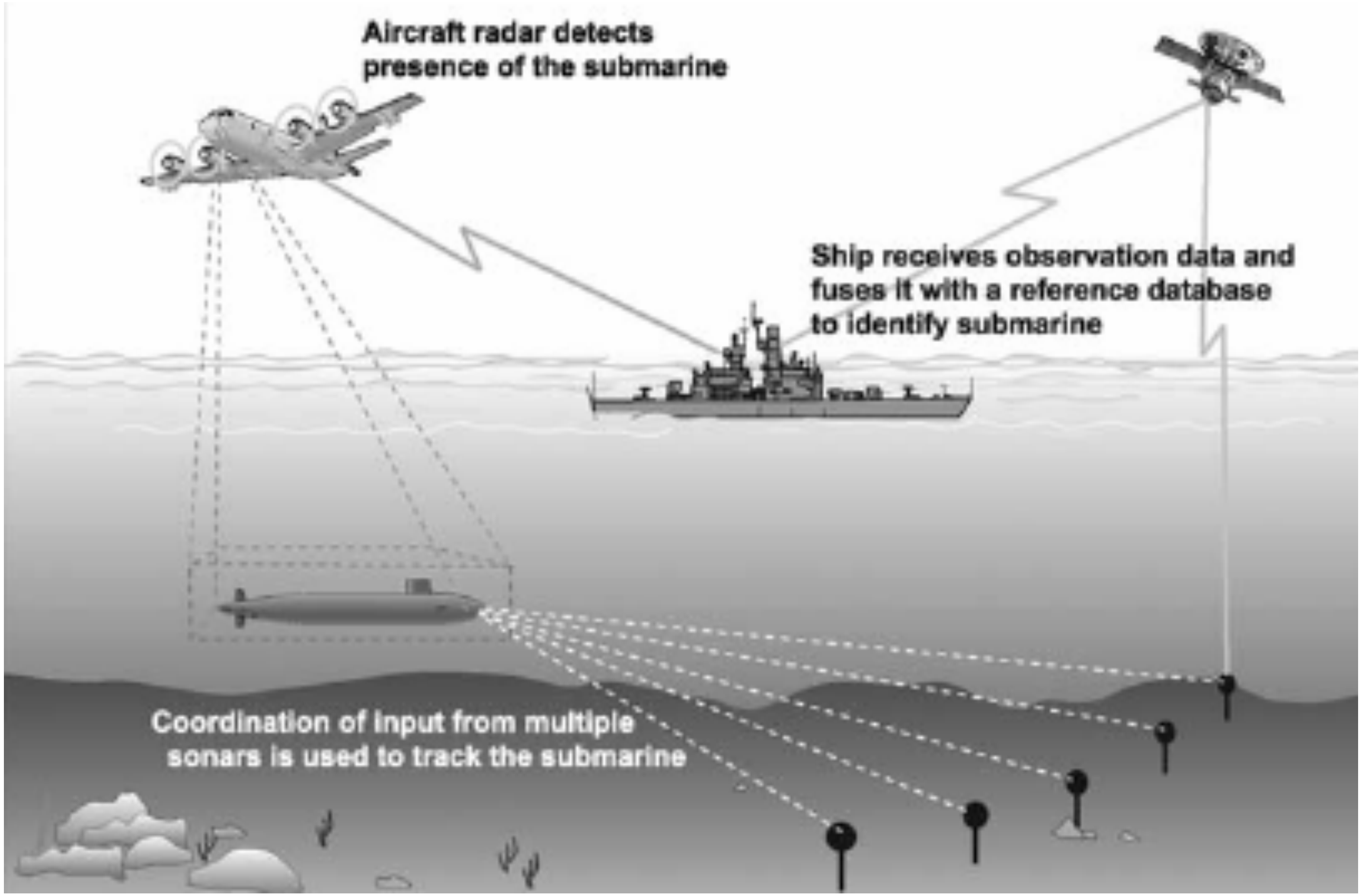
- Location and characterization of enemy units and weapons
- High level inferences about enemy situation
  - Relationship among units

Examples:

- Air to air or surface to air defense
- Ocean surveillance
- Battlefield intelligence

Note: mainly applied for target tracking and identification but limited works on high level reasoning

Anti-submarine Warfare



Civil Applications

- Fault diagnosis of manufacturing system
- Robotics

  - location and recognition of objects
  - guide the locomotion
- Medical diagnosis

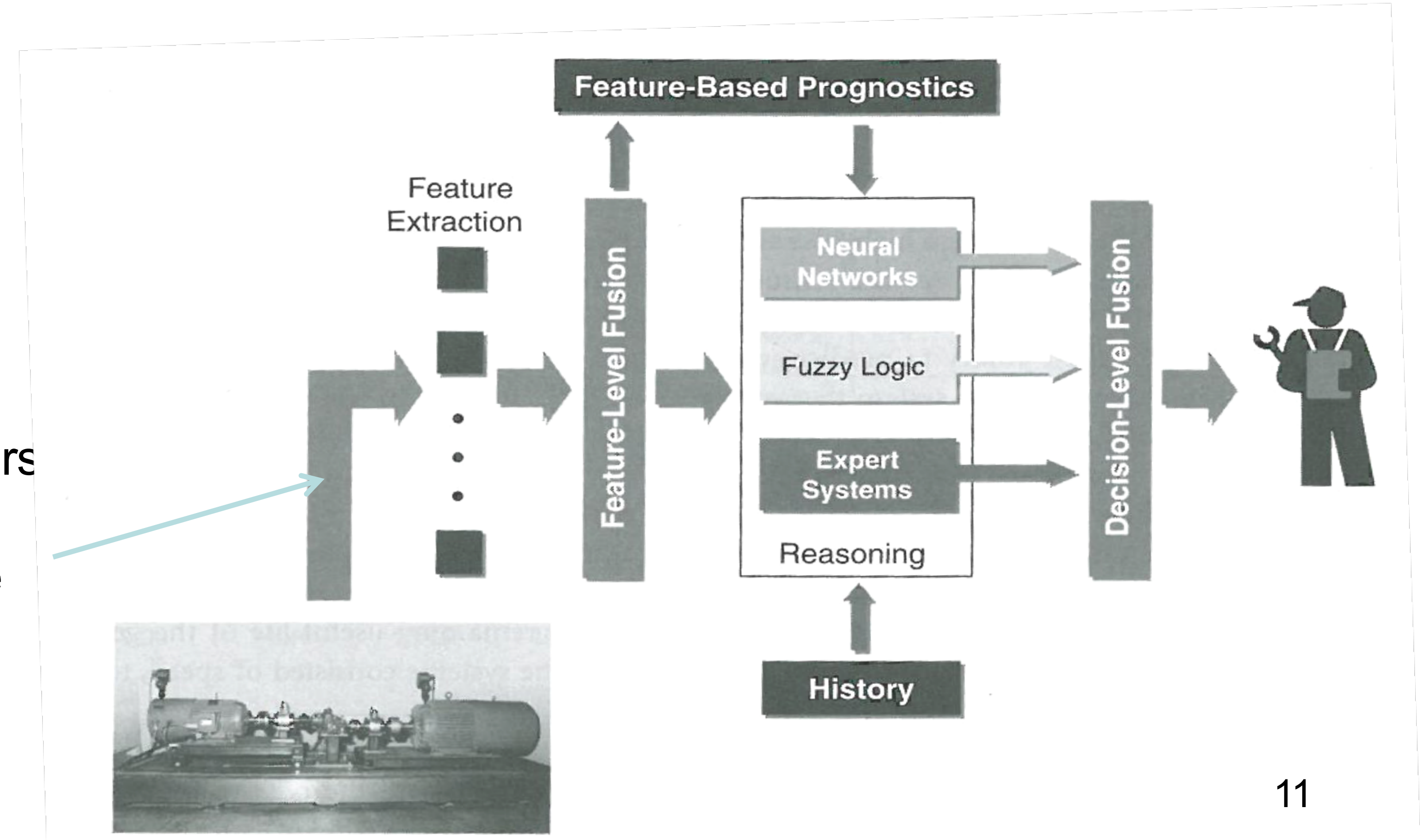
  - identification of abnormality or disease
- Environment monitoring

  - Identification and location of natural resources,
  - monitor natural disaster

Condition-Based Maintenance

- CBM has the aim to perform maintenance only when there is objective evidence of need or impending failure
- CBM involves predictive diagnostics depending on multi-sensor data
- Large cost saving by CBM compared with corrective maintenance and preventive maintenance

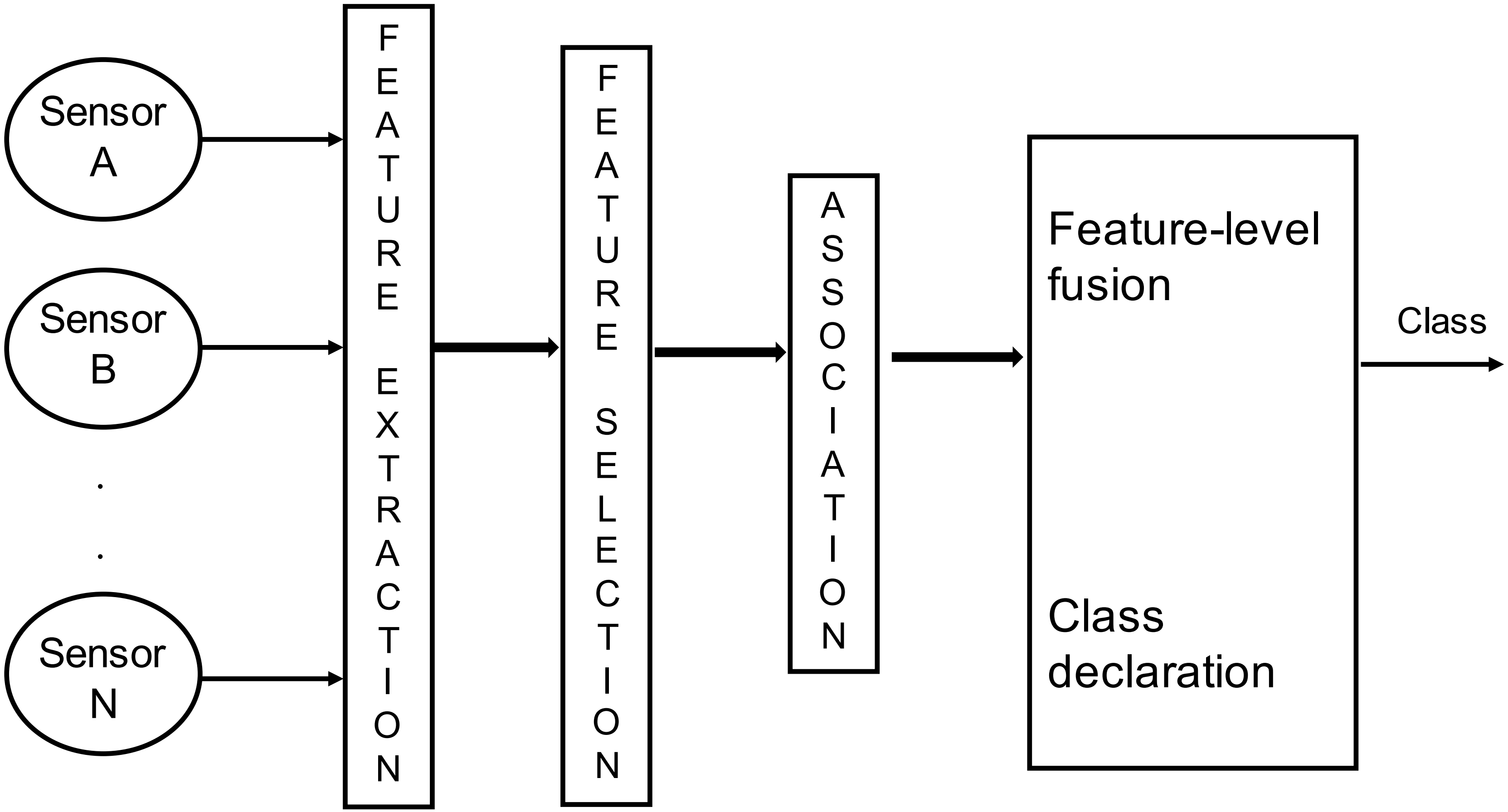
accelerometers
acoustic
thermocouple
oil quality



Mechanical Diagnostic Test Bed (MDTB)

Data processing architectures and related techniques

Data Processing Architecture 1



Features: time-domain attributes (rise time, fall time, peaks, mean, etc)
Fourier/wavelet coefficients, attributes from images, …

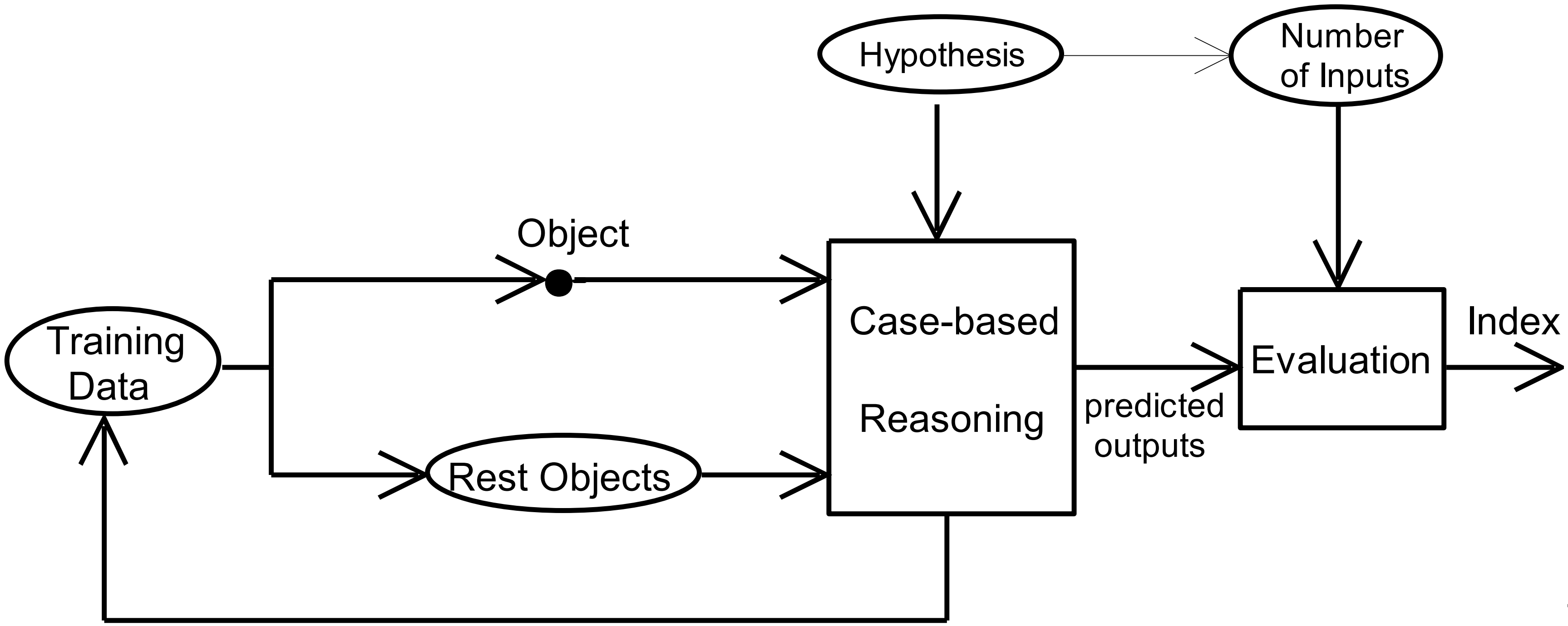
Class estimation: decision trees, support vector machines, neural networks, knowledge-based systems, fuzzy model , case-based reasoning, …..

Feature Selection: A Useful Extra Step

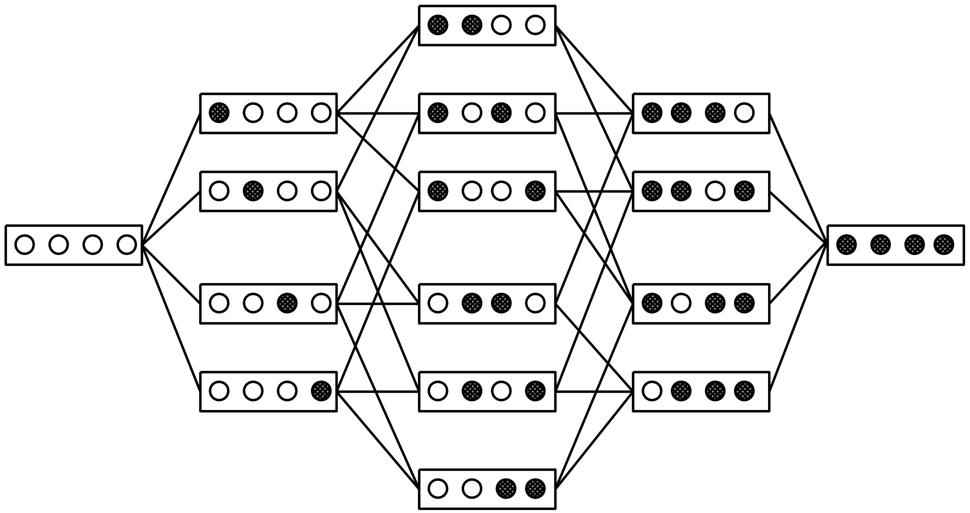
Aim: identify really relevant features to reduce the input dimension

A selected feature subset $\longleftrightarrow$ Hypothesis

Hypothesis evaluation by Means of Case-Based Reasoning (lazy learning)



Feature Selection: A Search problem

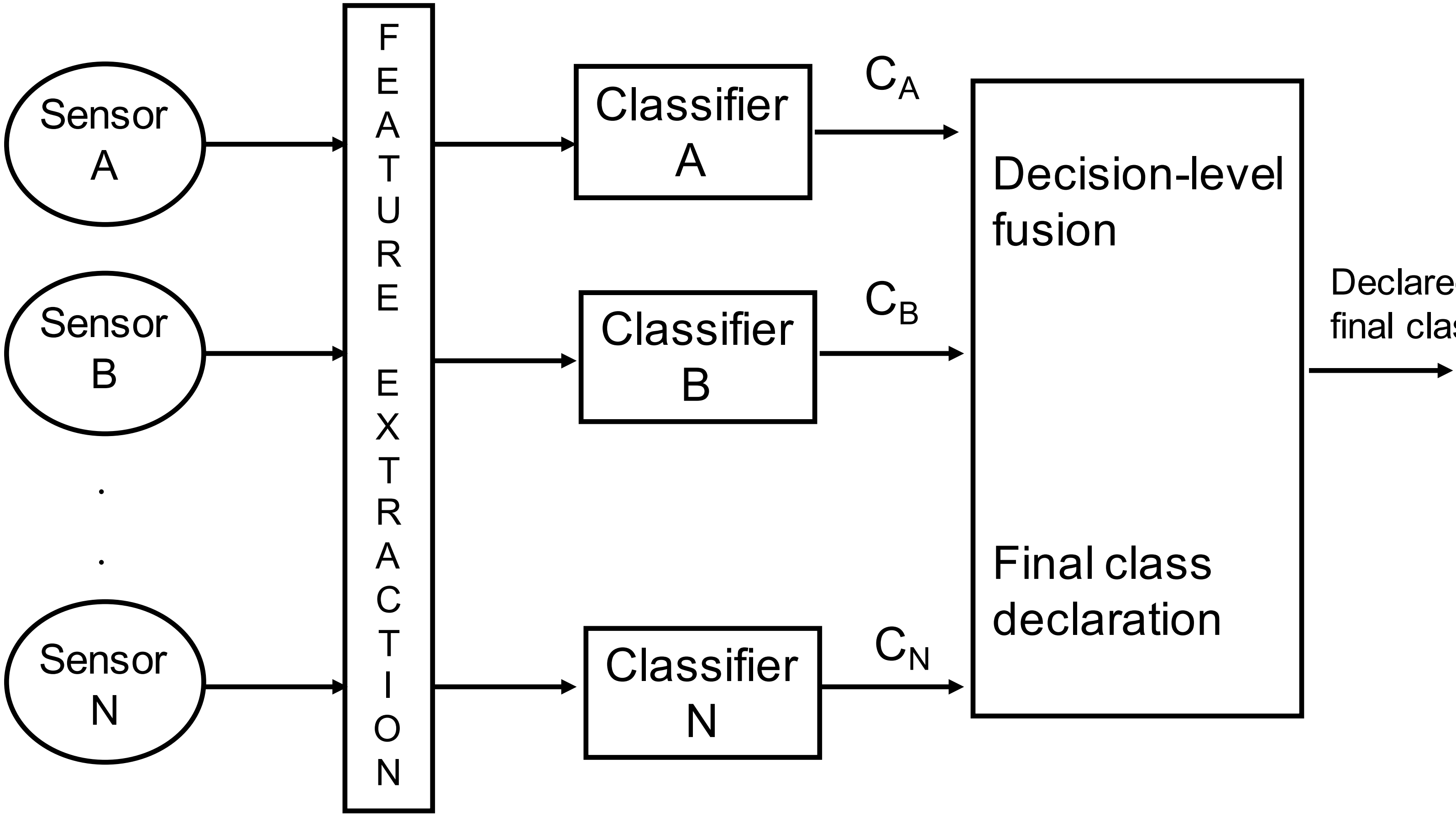


State (node): feature subset
Operator: connection between states

- Hill-climbing algorithm: choice of the best child node for the current node
- Best-first algorithm: choice of the best leaf node generated so far

N. Xiong, and P. Funk, "Construction of fuzzy knowledge bases incorporating feature selection", *Soft Computing*, Vol. 10, No. 9, 2006, pp. 796 – 804.

Data Processing Architecture 2



- Process each sensor data to conduct inference about estimated class
- Subsequent decision fusion to reach final decision

Decision-Level Fusion

1. Voting: decide the fused output decision based on the majority rule

Let X_i be a binary vector representing the prediction from sensor i with X_{ij}= 1 denoting the class j is selected and otherwise not selected.

The total votes won by class j is given by

$$y(j) = \sum_{i=1}^{N} X_{ij}$$

The final decision is the class having the most votes:

$$d = argmax_j y(j)$$

Decision-Level Fusion

2. Weighted voting: decide the fused output decision by weighting the predictions from individual sensors

Let X_i be a binary vector representing the prediction from sensor i with X_{ij}= 1 denoting the class j is selected and otherwise not selected. Let W_i be the weight for senor i with indication of the reliability of its classifier

The total weighted votes won by class j is given by

$$y(j) = \sum_{i=1}^{N} W_i X_{ij}$$

The final decision is the class having the most weighted votes:

$$d = argmax_j\, y(j)$$

Decision-Level Fusion

3. Bayesian Inference:

Let $P(H_j)$ be the prior probability for the truth of hypothesis H_j. Suppose a set of predictions $D_1, \ldots D_N$ have been made from data of sensors 1, 2, …N respectively. We use D_i as evidence to get posterior probabilities of hypotheses.

$$P(H_j|D_1,\ldots,D_N) = \frac{P(H_j)P(D_1,\ldots D_N|H_j)}{\sum_k P(D_1,\ldots D_N|H_k)P(H_k)} = \frac{P(H_j)\prod_{i=1}^{N} P(D_i|H_j)}{\sum_k P(H_k)\prod_{i=1}^{N} P(D_i|H_k)}$$

The final decision is the class having the most posterior probability:

$$d = argmax_j P(H_j|D_1,\ldots D_N)$$

Example: Bayesian Inference

$P(H_1)=P(H_2)=0.5$

Classifier from sensor 1

	H_1	H_2
D_{11}	0.8	0.1
D_{12}	0.2	0.9

Classifier from sensor 2

	H_1	H_2
D_{21}	0.7	0.4
D_{22}	0.3	0.6

Now the prediction from sensor 1 is class C_1 and the prediction from sensor 2 is class C_2, how can you update the probabilities for hypotheses?

Example: Bayesian Inference

Classifier from sensor 1

	H_1	H_2
D_{11}	0.8	0.1
D_{12}	0.2	0.9

Classifier from sensor 2

	H_1	H_2
D_{21}	0.7	0.4
D_{22}	0.3	0.6

Now given evidences D_{11} (C_1 predicted from sensor 1) and D_{22} (C_2 predicted from sensor 2), the probabilities for hypotheses are updated as follows:

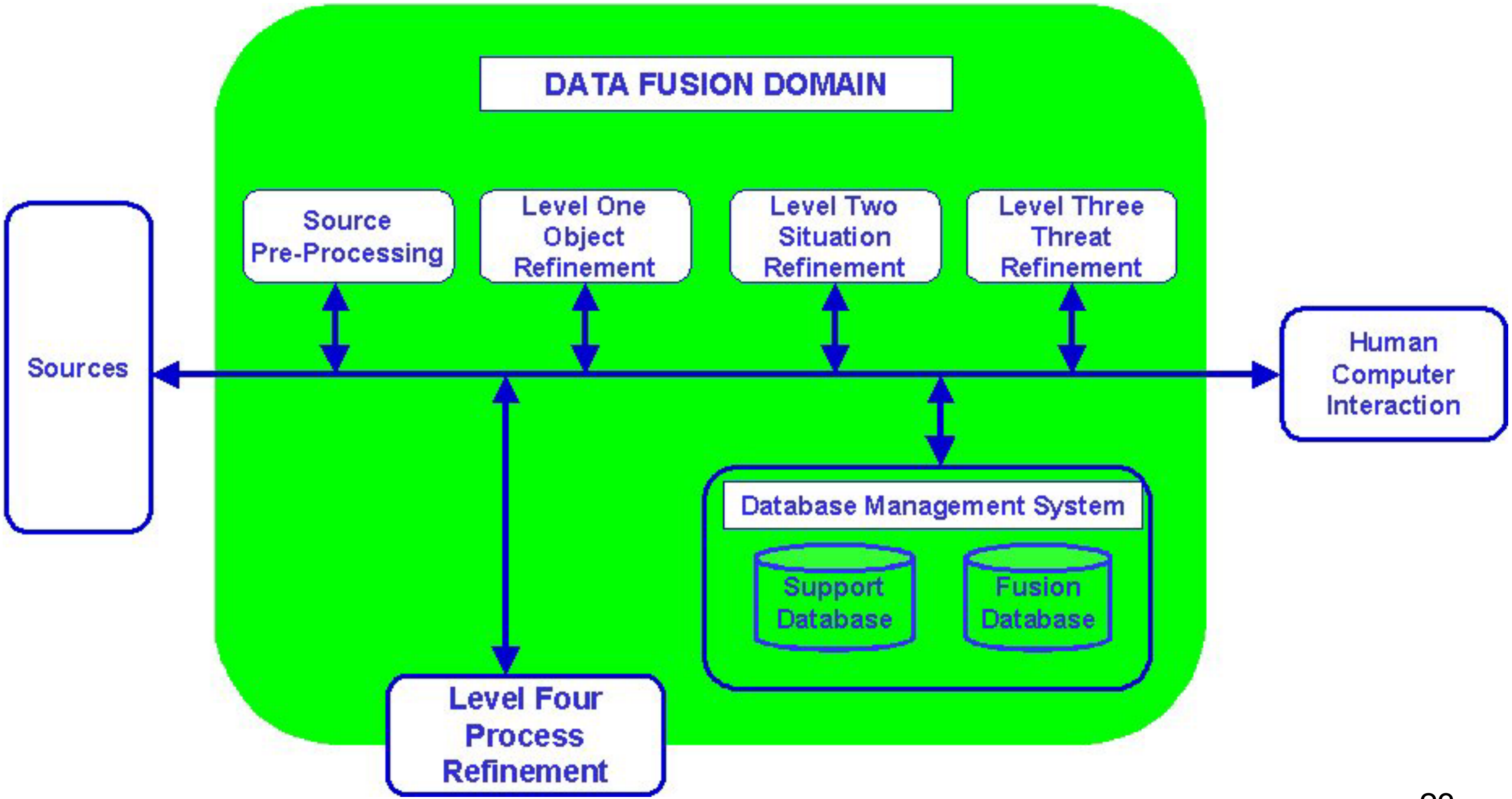
$$P(H_1|D_{11},D_{22}) = \frac{P(H_1)P(D_{11},D_{22}|H_1)}{P(D_{11},D_{22}|H_1)P(H_1)+P(D_{11},D_{22}|H_2)P(H_2)}$$

$$= \frac{P(H_1)P(D_{11}|H_1)P(D_{22}|H_1)}{P(D_{11}|H_1)P(D_{22}|H_1)P(H_1)+P(D_{11}|H_2)P(D_{22}|H_2)P(H_2)}$$

$$= \frac{0.5\times 0.8\times 0.3}{0.5\times 0.8\times 0.3+0.5\times 0.1\times 0.6} = 0.8$$

Fusion functionality in JDL model

JDL Data Fusion Model



Levels of Functions

- Level 1 (Object refinement): using sensor data to estimate an entity's position, velocity, attributes, and identity
- Level 2 (situation refinement): developing a description of the relation among entities and events in the environment
- Level 3 (Threat refinement): drawing inference to project current situation into future, to analyze enemy threat and opportunity of operations
- Level 4 (Process refinement): assessing and adjusting the overall data fusion process to improve (data fusion) system performance

Level 1: Object Refinement

- Most mature area in data fusion
- Estimate entity state (e.g. target tracking) by sequential usage of Kalman filter
- Kalman filter assumes linear world model and Gaussian noise

$$x(k) = Fx(k-1) + w(k-1)$$

$$y(k) = H\, x(k) + v(k)$$

- Inherently Kalman filter is based on the Bayes theorem of the continuous form, and works with the steps: prediction → update

Particle Filter

- No restrictive assumption on the world model and characteristics of noise
- Not rely on mathematical ways to express the probability distribution the estimated states
- Use a set of samples to approximate the (hidden) probability distribution of the state
- Use the Bayes theorem to revise the set of samples in terms of observations to acquire more accurate state estimations.

Particle Filter

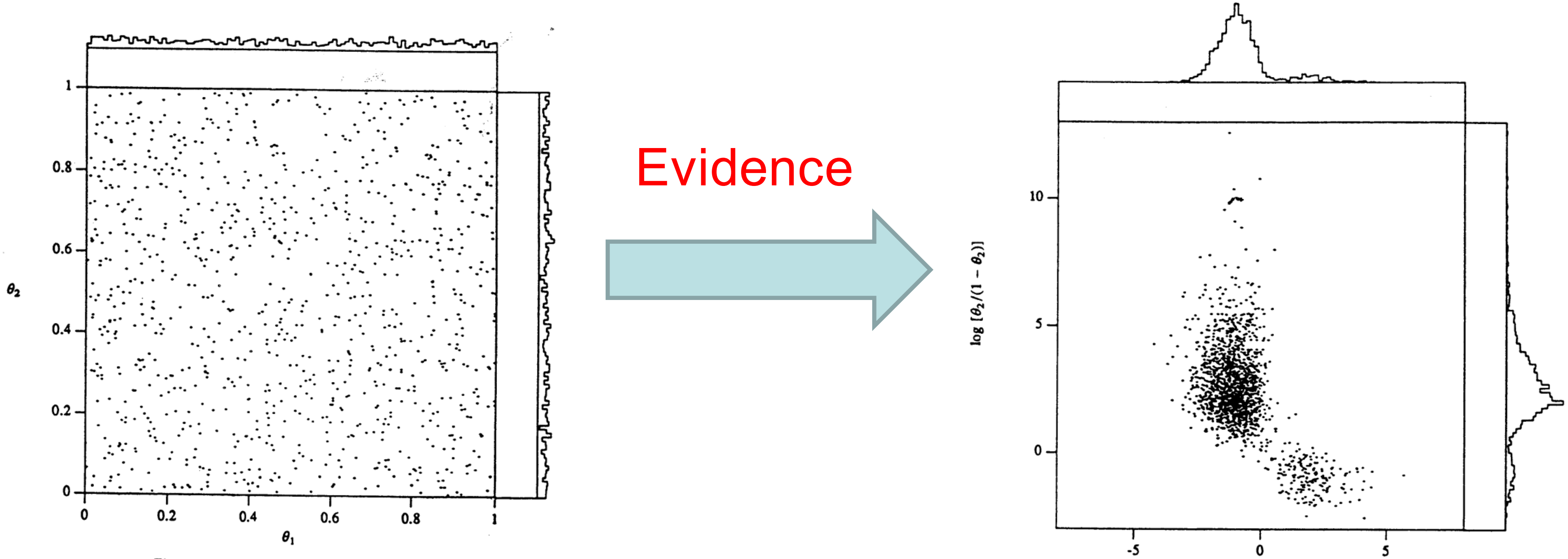


Figure 1. Sample from Prior.

Figure 3. Sample from Transformed Posterior.

Interested students can read the paper “Bayesian statistics without tears: a sampling and resampling perspective”, The American Statistician, Vol. 46, No. 2, 1992.

Level 2 and 3 Functions

- Relatively immature research areas, few robust systems

- Current works dominated knowledge-based methods and rule-based systems

- The main challenge is construction of viable knowledge bases to support such high level reasoning about situation and threat assessment

- Some kind of learning is also needed to induce relationships among entities from observations.

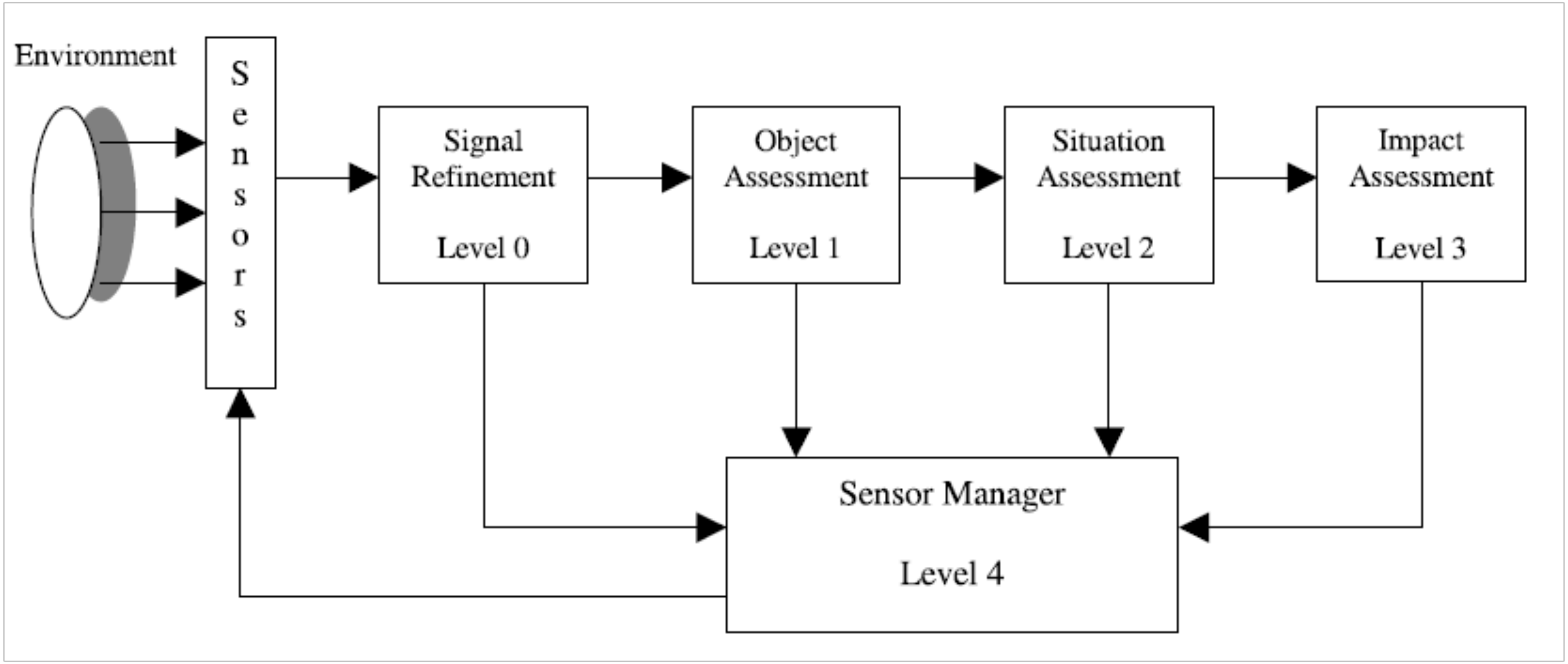
Level 4: Process Refinement

•In principle this level includes both adaptation of data fusion algorithms/methods and management of data collection resources (sensors) to improve the overall data fusion system performance

• To date, the works on level 4 are dominated by sensor management, which aims to coordinate the usage of a suite of sensors to enhance fusion effect.

Sensor Management

- A comprehensive survey "Multi-sensor management for information fusion – issues and approaches", by Ning Xiong & Per Svensson, Information Fusion 3 (2002), pp. 163 - 186

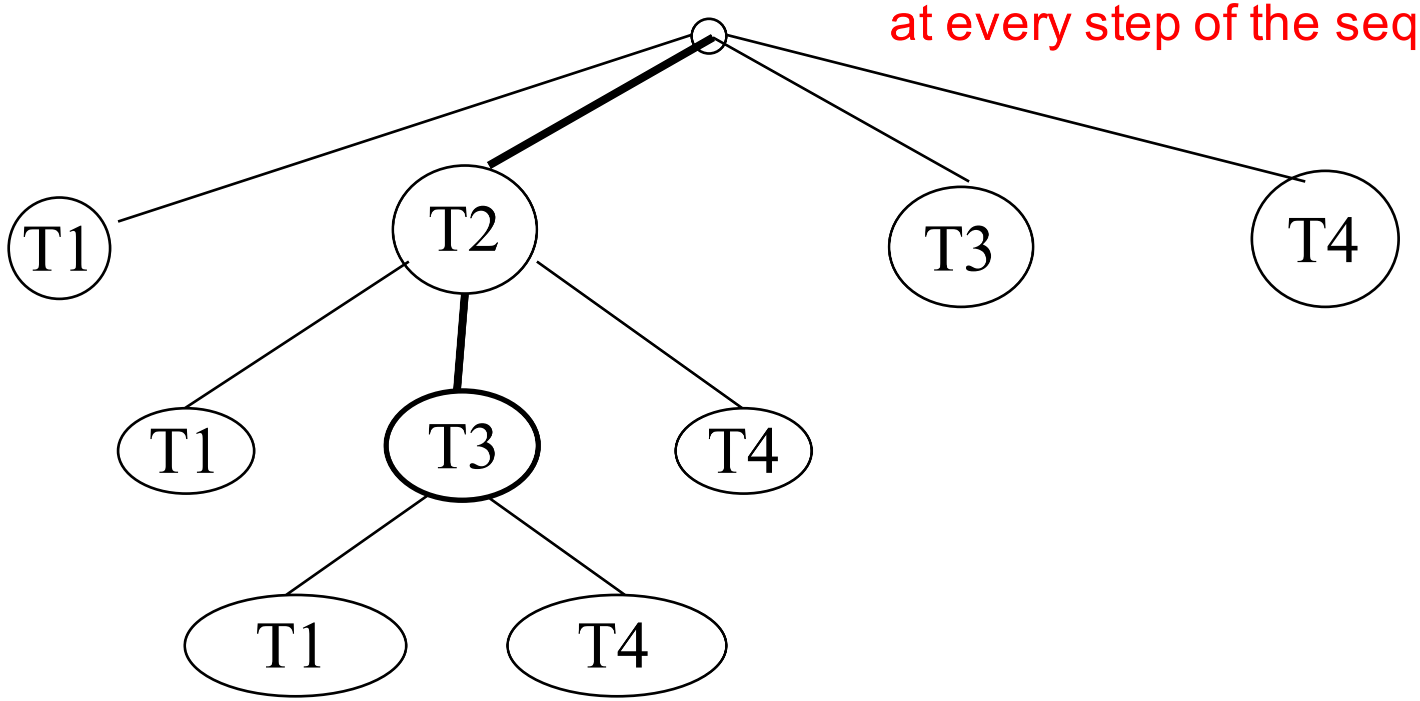


Closed loop and feedback structure

Decision Theoretic Sensing Action Selection

Using Bayesian decision theory to analyze whether sensing Action is worthy or which sensing action gives the most value of information for decision making

Choosing the best sensing action at every step of the sequence



Recommendations for Reading

1. Read and understand the contents of the slides given in the lecture.

2. Read the article "Multisensor data fusion" by David Hall, available in blackboard.

3. Interested students can also read the papers on Sensor Management and/or Particle Filter, which are available in the blackboard. But they are not obligatory.