Mode Estimation of Probabilistic Hybrid Systems

Michael Hofbaur[1&2] & Brian C. Williams[1]

[1)] Artificial Intelligence & Space Systems Laboratories
MIT, USA

[2)] Department of Automatic Control,
TU-Graz, Austria

Motivation

Advanced Life Support System
-
BIO-Plex

- Highly complex artifact
- Long autonomous operation in a harsh environment
- Robust operation – fault tolerance

Monitoring and diagnosis capabilities are critical for building highly autonomous artifacts that can operate robustly in harsh environments of a long period of time.

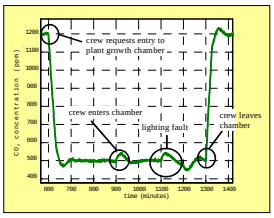
Overview

- Probabilistic Hybrid Automata
  – *Model & Execution*
- Concurrent Probabilistic Hybrid Automata
- Hybrid Estimation
  – *Overview – intuitively & filtering background*
  – *Problem Formulation*
  – *A* Formulation*
- Example
- Discussion & Conclusion

why Hybrid Mode/State Estimation?

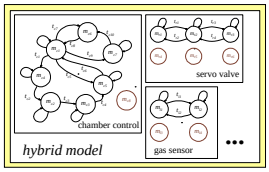
Monitoring and Diagnosis has to track the system's behavior along both its *continuous state changes* and its *discrete mode changes* and their system-wide interaction.

- *operational* modes
- *failure* modes
- *estimation and filtering* of continuously valued variables



Hybrid Model

concurrent Probabilistic Hybrid Automata (cPHA)



A concurrent Probabilistic Hybrid Automata (cPHA) is a hidden Markov model, encoded as a set of components with modes that exhibit a continuously valued dynamical behavior that is expressed by difference / algebraic equations.

Probabilistic Hybrid Automata

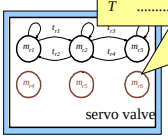
Probabilistic Hybrid Automata $\langle \mathbf{x}, \mathbf{w}, F, T, X_d, U_d, T_s \rangle$

$\mathbf{x} \ldots \{x_d\} \text{È} \mathbf{x}_c$ ..... x_d mode (discrete state) with domain X_d
$\mathbf{x}_c$ continuous state with domain $\mathbb{R}^n$

$\mathbf{w} \ldots \mathbf{u}_d \text{È} \mathbf{u}_c \text{È} \mathbf{y}_c \ldots$ $\mathbf{u}_d$ discrete command with domain U_d
$\mathbf{u}_c$ continuous command with domain $\mathbb{R}^{m_i}$
$\mathbf{y}_c$ continuous output with domain $\mathbb{R}^{m_o}$

F ..... discrete-time dynamics for each mode (sampling-period T_s)

T ..... guarded probabilistic transitions between modes



Mode / State Transition

Discrete mode changes and continuously valued evolution of the state variables take place at two different rates:

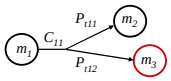
a) continuous evolution is captured at the sampling-rate

$$\mathbf{x}_{c,(k)} = f\left(\mathbf{x}_{c,(k-1)}, \mathbf{u}_{c,(k-1)}, x'_{d,(k-1)}\right), \mathbf{y}_{c,(k)} = g(\mathbf{x}_{c,(k)}, \mathbf{u}_{c,(k)}, x_{d,(k)}),$$

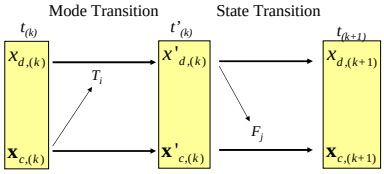
$$t_{(k)} = t_{(k-1)} + T_s$$

b) probabilistic mode changes take place instantly

$$\left\langle x_{d,(k)}, \mathbf{x}_{c,(k)} \right\rangle \xrightarrow{guard(\mathbf{x}_{c,(k)}, \mathbf{u}_{c,(k)})} \left\langle x'_{d,(k)}, \mathbf{x}'_{c,(k)} \right\rangle$$



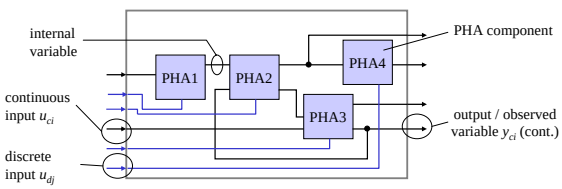
Mode / State Transition



Mode transition: time proceeds only infinitesimally $t'_{(k)} = t_{(k)} + \varepsilon$ so that the evolution of the continuous state $\mathbf{x}_{c,(k)} \rightarrow \mathbf{x}'_{c,(k)}$ can be neglected: $\mathbf{x}'_{c,(k)} = \mathbf{x}_{c,(k)}$

State transition: no transition is triggered ($x'_{d,(k)} = x_{d,(k+1)}$) and time proceeds for one sampling period: $t_{(k+1)} = t_{(k)} + T_s$. The evolution of the continuous state $\mathbf{x}'_{c,(k)} \rightarrow \mathbf{x}_{c,(k+1)}$ is captured by the discrete-time dynamic model that holds for $x'_{d,(k)}$.

concurrent Probabilistic Hybrid Automata

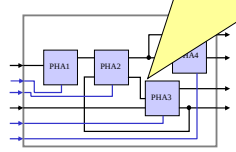


- concurrent PHA components are connected to inputs (continuous and discrete) and outputs of the cPHA and interconnected by internal variables.
- observed variables = internal variable + additive Gaussian noise

Probabilistic Hybrid Automata

Concurrent Probabilistic Hybrid Automata $\left\langle A, \mathbf{u}, \mathbf{y}_c, \mathbf{v}_s, \mathbf{v}_o, N_x, N_y \right\rangle$

A	set of PHAs
$\mathbf{u} \ldots \mathbf{u}_d$ È $\mathbf{u}_c \ldots$	continuous and discrete command variables
$\mathbf{y}_c$	observed continuous variables
$\mathbf{v}_s, \mathbf{v}_o$	state disturbances and sensor noise inputs characterized by N_x, N_y



$$\mathbf{x}_c = \mathbf{x}_{c1} \text{ È } \mathbf{x}_{c2} \text{ È} \ldots \text{È } \mathbf{x}_{cl}$$

$$\mathbf{x}_d = \{x_{d1}, x_{d2}, \ldots, x_{dl}\}$$

$$\mathbf{x}_{c,(k)} = f\left(\mathbf{x}_{c,(k-1)}, \mathbf{u}_{c,(k-1)}, \mathbf{x}'_{d,(k-1)}\right) + \mathbf{v}_{s,(k-1)}$$

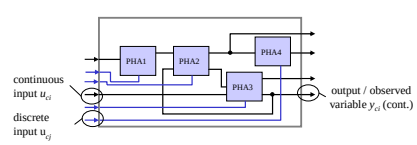
$$\mathbf{y}_{(k)} = g(\mathbf{x}_{c,(k)}, \mathbf{u}_{c,(k)}, \mathbf{x}_{d,(k)}) + \mathbf{v}_{o,(k)}$$

Roadmap

- Probabilistic Hybrid Automata
  - *Model & Execution*
- Concurrent Probabilistic Hybrid Automata
- Hybrid Estimation
  - *Overview – intuitively & filtering background*
  - *Problem Formulation*
  - *A* Formulation*
- Example
- Discussion & Conclusion

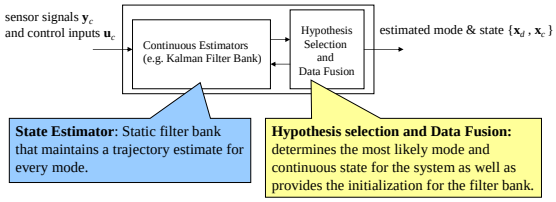
Hybrid Mode / State Estimation

Task Overview:



Hybrid Estimation Problem: Given a cPHA model for a system, a sequence of observations and the history of the control inputs generate *the leading set of most likely states at time-step k*

Background: Multi-Model Estimation



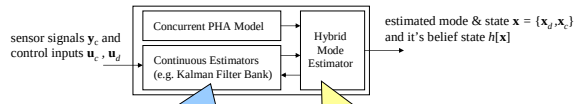
State Estimator: Static filter bank that maintains a trajectory estimate for every mode.

Hypothesis selection and Data Fusion: determines the most likely mode and continuous state for the system as well as provides the initialization for the filter bank.

advantages: high fidelity estimation of continuous behaviors
noise handling and incipient fault detection

disadvantages: limited to tracking a small number of hypothesis
(limited size of the filter bank)

hybrid Mode / State Estimation



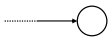
Hybrid State Estimator
Maintains the set of most likely hybrid state estimates as a set of trajectories.
A Hidden Markov Model style belief state update is used to determine the likelihood for each traced trajectory

Hybrid Mode estimator:
Determines for each trajectory the possible transitions, and specifies (dynamically) the candidate trajectories to be tracked by the continuous state estimators.

hybrid Mode/State Estimation

At each time step k, we evaluate for each trajectory:

old estimate:
$\mathbf{x}_{(k-1)} = \{\mathbf{x}_{d,(k-1)}, \mathbf{x}_{c,(k-1)}\}$, $h_{(k-1)}$

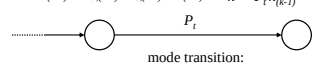


hybrid Mode Estimation

At each time step k, we evaluate for each trajectory:

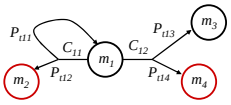
old estimate:
$\mathbf{x}_{(k-1)} = \{\mathbf{x}_{d,(k-1)}, \mathbf{x}_{c,(k-1)}\}$, $h_{(k-1)}$

$\mathbf{x}'_{(k-1)} = \{\mathbf{x}'_{d,(k-1)}, \mathbf{x}_{c,(k-1)}\}$,
$h' = P_t\, h_{(k-1)}$



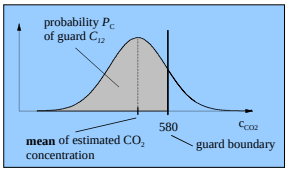
mode transition:
$\mathbf{x}_{d,(k-1)} = \mathbf{m}_i \rightarrow \mathbf{x}'_{d,(k-1)} = \mathbf{m}_j$

Transition Probability P_o



C_{12} guards the transition to either m_3 (*nominal* transition) or to m_4 (*failure* transition):

$$C_{12} : c_{CO2} \text{ £ } 580\, ppm$$



*transition probability = guard probability * thread probability*

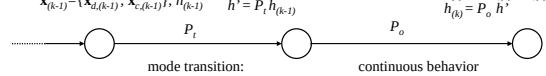
hybrid Mode Estimation

At each time step k, we evaluate for each trajectory:

old estimate:
$\mathbf{x}_{(k-1)} = \{\mathbf{x}_{d,(k-1)}, \mathbf{x}_{c,(k-1)}\}$, $h_{(k-1)}$

$\mathbf{x}'_{(k-1)} = \{\mathbf{x}'_{d,(k-1)}, \mathbf{x}_{c,(k-1)}\}$,
$h' = P_t\, h_{(k-1)}$

new estimate
$\mathbf{x}_{(k)} = \{\mathbf{x}_{d,(k)}, \mathbf{x}_{c,(k)}\}$,
$h_{(k)} = P_o\, h'$



mode transition:
$\mathbf{x}_{d,(k-1)} = \mathbf{m}_i \rightarrow \mathbf{x}'_{d,(k-1)} = \mathbf{m}_j$

continuous behavior
$\mathbf{x}'_{c,(k-1)} \rightarrow \mathbf{x}_{c,(k)}, \mathbf{x}_{d,(k)} = \mathbf{x}'_{d,(k-1)}$

Observation Probability P_t

We compare the sensor signal $\mathbf{y}_{c(k)}$ with its estimation for mode $\mathbf{m}_j$ using an extended Kalman filter.

operation performed by an (extended) Kalman filter:

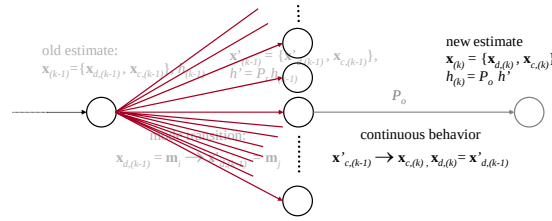
1) state prediction: $\mathbf{x}_{c,(k-1)}, \mathbf{P}_{(k-1)}, \mathbf{u}_{c,(k-1)} \rightarrow \mathbf{x}'_{c,(k)}, \mathbf{P}'_{(k)}$
- residual calculation: $\mathbf{x}'_{c,(k)}, \mathbf{P}'_{(k)}, \mathbf{y}_{c(k)} \rightarrow \mathbf{r}_{(k)}, \mathbf{S}_{(k)}, P_o$
- Kalman filter gain calculation: $\mathbf{P}'_{(k)} \rightarrow \mathbf{k}_{(k)}$
- state estimate refinement: $\mathbf{x}'_{c,(k)}, \mathbf{P}'_{(k)}, \mathbf{k}_{(k)}, \mathbf{r}_{(k)} \rightarrow \mathbf{x}_{c,(k)}, \mathbf{P}_{(k)}$

$$P_o = e^{-\mathbf{r}^T \mathbf{S}^{-1} \mathbf{r}}$$

→ *one extended Kalman filter for each hypothesis*

exponential Explosion

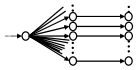
At each time step k, we evaluate for each trajectory:



The number of possible transitions at each time step can be very large:
E.g. a model with 10 components, each of which can transition to 3 successor modes has $3^{10} = 59049$ possible successor modes *for each trajectory at each time step*!

Approach:

how do we tackle the exponential blowup?

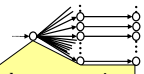


- A mode transition of a component is guarded by $C(\mathbf{x}_{c,(k)}, \mathbf{u}_{d,(k)})$ i.e. it depends on the continuous state, local to the component, and the component's discrete command input.
- no discrete interconnection among the components

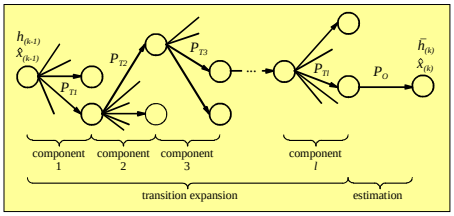
→ component transitions independent of each other
→ take mode transition of the cPHA component-wise

Approach:

formulate node expansion as:
A* search:

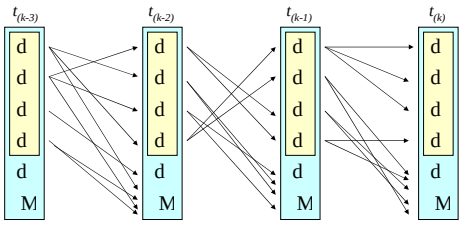


Transition Expansion – expand component-wise:

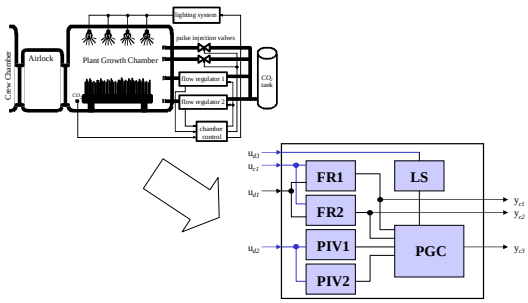


Overall hybrid Mode Estimation Scheme:

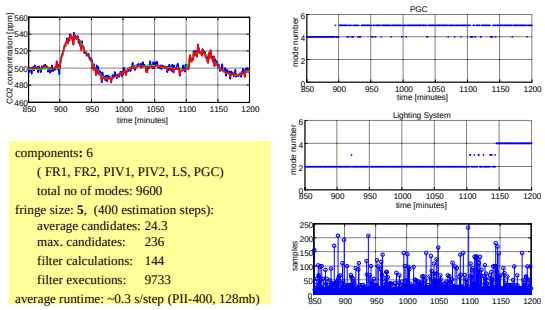
formulate hybrid estimation as: **beam search:**



Bio-Plex / Plant Growth Chamber

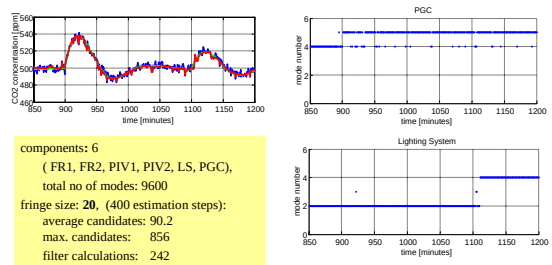


Simulation Result



components: 6
(FR1, FR2, PIV1, PIV2, LS, PGC)
total no of modes: 9600
fringe size: **5**, (400 estimation steps):
average candidates: 24.3
max. candidates: 236
filter calculations: 144
filter executions: 9733
average runtime: ~0.3 s/step (PII-400, 128mb)

Simulation Result



components: 6
(FR1, FR2, PIV1, PIV2, LS, PGC),
total no of modes: 9600
fringe size: **20**, (400 estimation steps):
average candidates: 90.2
max. candidates: 856
filter calculations: 242
filter executions: 36050
average runtime: ~1 s/step (PII-400, 128mb)

Discussion & Conclusion

Summary

- cPHA - discrete-time dynamics, probabilistic transition
- additive Gaussian noise model - extended Kalman filter
- comparison to multi-model filtering

Future Research

- other noise models - particle filters
- unknown mode - decomposition
- conflict directed search
- hME in the context of generalized state feedback control
- mode estimation - mode reconfiguration

optional slides

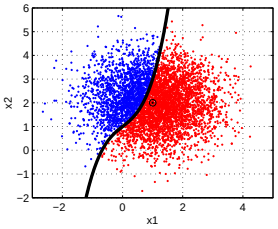
Guard Probability P_{co} multi-var. -case

e.g. 2nd order system

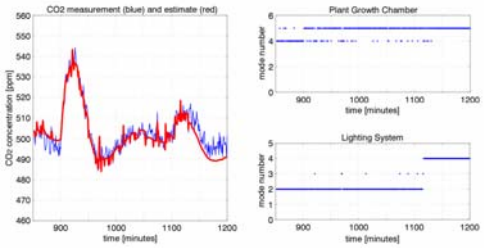
$$\mathbf{x} = \begin{bmatrix} d \\ e \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} d & 0 \\ e & 1 \end{bmatrix}$$

guard: $1 + x_1 + x_1^3 - x_2 < 0$

® $P_{co} = \dfrac{5157}{15000} = 0.344$



Simulation Result



components: 4 (flow regulator, pulse valve, light system, chamber), total no of modes: 480
fringe size: 20, average candidates: 57, filter calculations (total for experiment): 60
average runtime 0.3 s/step (PII-400, 128mb), space used (total): 1MB (program & data)