Controlling Actuation in Central HVAC Systems in Buildings

Jason Koh¹, Bharathan Balaji¹, Rajesh Gupta¹, and Yuvraj Agarwal²
CSE, University of California, San Diego¹, CSE, Carnegie Mellon University²

Need for Control Platform

**Apps with fine-grained control are missing**

Need Abstraction of Control

We propose “Building Control Engine” to provide high level view of control.

**Design Goal**

1. Provide a library as API
2. Reject unsafe operation
   - Should not exceed operating range. e.g., Temperature Setpoint to 40°F
   - Should not operate too frequently. e.g., Damper command change every 30 seconds.
3. Status of operation per a sensor/actuator is stored/managed
   - Points are dependent on others.
   - Need to track status of each resource, and its dependency information.
4. Logging entire history of control
   - Logging history of all system status
   - Need for both data analysis and system management
5. Rollback
   - In case of system down or user interrupt.

**HVAC Background**

**Sensor/Actuator Dependency Graph**

Application Suggestion

1. System Diagnosis
   - Active fault diagnosis
   - System estimation for security
2. System Identification
   - Sensor/actuator colocation
   - Type identification
   - Finding control function
   - Finding dependency graph
3. Personalized Control

**Validation Algorithm**

1. Unsafe operation
   - Definition of unsafe operation
   - Exceeding normal operating range.
   - Too frequent operation.
   - Each actuator has its own definition of unsafety.
   - Each operation affects dependent points.

**Building Control Engine (BCE)**

**Challenging Scenarios**

**Example Application: Sensor/Actuator Co-location**

**Problem:** Co-location of sensor/actuators when location metadata is unavailable.

**Assumption:** Sensor and actuator types are known.

**Hypothesis:** Co-located points will be distinguishable if unique control signal is applied. The signal is unique in terms of amplitude, frequency, and phase.

**Method:** Use control to make information more observable.

**Algorithm for Co-location**

- Apply large pulses to a Temperature Setpoint
- Extract time-series features from each point over all zones

**Result:** 68% recall with 98.6% precision over 8 zones.