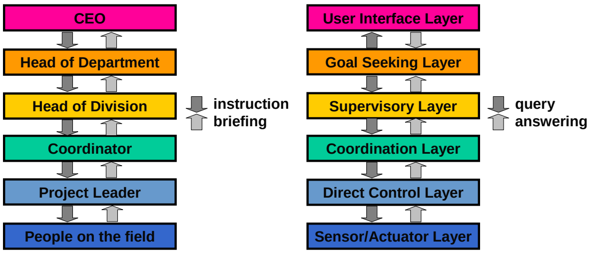
Layered Control Structure and Learning Algorithms

Authors, Prof. David Auslander, Prof. Ed Arens, Charlie Huizenga

Graduate Students : Jaehwi Jang
Therese Peffer, Xue Chen, Anna LaRue, and Carrie Brown
Mechanical Engineering, Architecture, UC Berkeley

Vision

DR thermostat control is characterized by a layered structure and self-learning. These two features will make it possible to handle complexity efficiently in controlling residential buildings with various characteristics



Hierarchical Structure in a Company and Control Code

Why is a layered structure beneficial?
Layering provides modularity and interoperability to the control structure.
modularity: design teams work on a layer
interoperability: modules from various sources can interact
However, it has some disadvantages as well.
local optimization: optimization is limited in a layer
limitation on adding functionality: due to fixed interfaces between layers, adding functionality requiring new information is hard

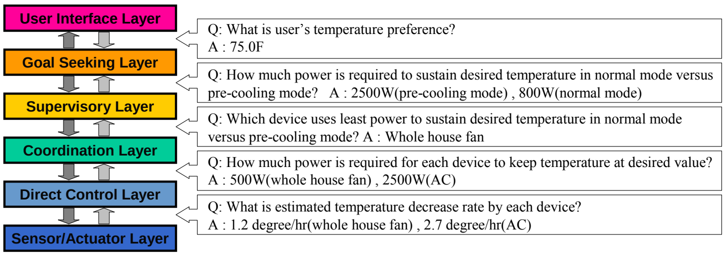
Why is learning crucial for thermostat control in residential buildings?
The most distinctive difference between commercial buildings and residential buildings is existence of professional management. Residential DR systems does not have professional management. The unique solution is to develop autonomous systems which works well in any types of residential buildings. Learning will provide the following features.

- reacting reasonably to a price or reliability signal even if nobody is home
- working satisfactorily out-of-the-box

Methods

Function of each layer

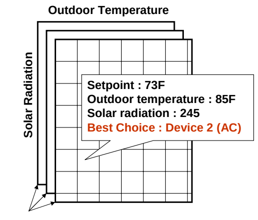
- User Interface Layer: obtain user preference
- Goal Seeking Layer: perform decision making (choose best mode in a given condition)
- Supervisory Layer: deal with optimal temperature (setpoint) for each mode
- Coordination Layer: choose optimal (least power consumed) device in a given condition
- Direct Control Layer: apply feedback control, PID control, or PWM control, etc.
- Sensor/Actuator Layer: deal with HVAC device information, house parameter identification



Example of Query in Normal Operation

What to learn / How to learn

- Supervisory Layer: optimal setpoint in a given condition
- Coordination Layer: least power consumed device to fulfill requirement
- Direct Control Layer: feedback gain, power requirement of each HVAC device
- Sensor/Actuator Layer: house parameter, occupancy pattern



Look-up Table Method

Simplified heat flow model:
(Δ indoor temperature)/(Δ time)
= $a + b$(outdoor temp – indoor temp) + c(radiation)
$z = a + bx + cy$

Least square error:
$\Pi = \sum[z_i - (a + bx_i + cy_i)]^2$

parameter a, b, and c should minimize least square error
$\sum z_i = a\sum 1 + b\sum x_i + c\sum y_i$
$\sum x_i z_i = a\sum x_i + b\sum x_i^2 + c\sum x_i y_i$
$\sum y_i z_i = a\sum y_i + b\sum x_i y_i + c\sum y_i^2$

Least Square Method

Research Questions

In current step, three main problems occur. The first problem is what each layer should do. It is about functions for layers. The second problem is what each layer should learn. Learning is performed by each layer not only for doing a layer's job but also for answering interacting layers' query. The third problem is how to learn it. This methodological problem can be the key to provide more efficient and accurate results.

How can the whole control system be hierarchically layered based on functionality?

- Defines function of each layer clearly
- Sets up the rule on information sharing and interchanging (sensor data, relay command, and modified data by a certain layer, etc.)
- Defines control information flow and query flow

What should each layer learn?

- Each layer needs to learn what it needs to perform its own function well
- Each layer needs to learn what it needs to respond to queries from other layers

What kind of learning algorithms can we use and how can we develop them for our purpose?

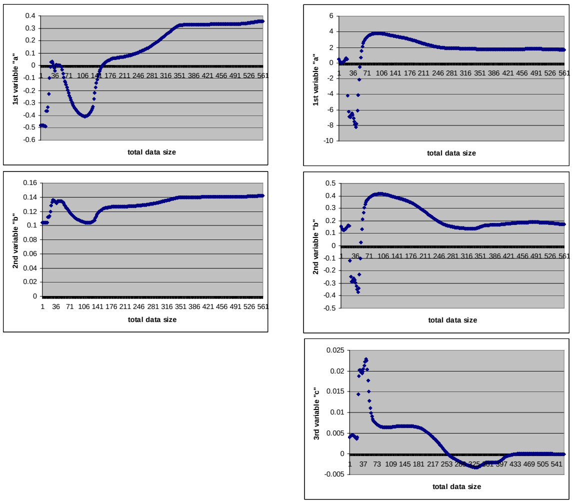
- Can apply regression (curve fitting) algorithm such as least square method for house parameter identification
- Can apply multi-dimensional look-up table method to for some learning parts

Findings

By using the simplified house model and least square method, house parameters which minimize least square error could be acquired.

- For simulation, the JAVA control code and the MZEST (Multi-Zone Energy Simulation Tool) were used
- Simulations were performed in two cases, night time period (solar radiation does not exist) and day time period (solar radiation exists)
- Assumed that house parameter a, b, and c are constant

Night Time Period | Day Time Period



- Each house parameter converges to a certain value
- In day time period, least square error was relatively large compared with that in night time period
- a and c can be functions of time, $a(t)$ and $c(t)$
- More complicated (realistic) house model will provide more reliable result

Multi-dimensional look-up table method can cause problems!

- How to update the table? (smarter way than just averaging?)
- More than two dimensional loop-up table may not be filled up with data even over long time period
- Restrict on the dimension of loop-up table
  – does this table include enough information for prediction?
- Keep the initial dimension (more than two) and fill it up with acquired data and correlation between data

DRETD
Demand Response Enabling Technology Development