A Learning Agent for Heat-Pump Thermostat Control

Daniel Urieli and Peter Stone
Department of Computer Science
The University of Texas at Austin
{urieli,pstone}@cs.utexas.edu
Chapter 1: Buildings Sector

WORLD ENERGY CONSUMPTION

- OTHER: 40%
- OECD: 15%
- EUROPE: 15%
- CHINA: 20%
- RUSSIA: 6%

U.S. ENERGY CONSUMPTION

- INDUSTRIAL: 30%
- RESIDENTIAL: 22%
- COMMERCIAL: 19%
- TRANSPORTATION: 29%
- OTHER: 19%

U.S. Department of Energy
Energy Efficiency & Renewable Energy
Chapter 2: Residential Sector

Residential Site Energy Consumption by End Use

- Space Heating: 45%
- Water Heating: 18%
- Space Cooling: 9%
- Lighting: 6%
- Cooking: 4%
- Electronics: 5%
- Refrigeration: 4%
- Wet Cleaning: 3%
- Computers: 2%
- Other: 3%
- Adjust to SEDS: 4%

Heating, Ventilation, and Air-conditioning (HVAC) systems
Heat-Pump based HVAC System

- Heat-pump is widely used and highly efficient
  - Its heat output is up to 3x-4x the energy it consumes
  - Consumes electricity (rather than gas/oil based)
    - can use renewable resources
  - But: no longer effective in freezing outdoor temperatures

- Backed up by an auxiliary heater
  - Resistive heat coil
  - Unaffected by outdoor temperatures
  - But: consumes 2x the energy consumed by the heat-pump heater

- Heat pump is also used for cooling
Thermostat – an HVAC System’s Decision Maker

• The thermostat:
  – Controls Comfort
  – Significantly affects energy consumption

• Current interest evident from the appearance of startup companies like NEST, as well thermostats by more traditional companies like Honeywell
Goal:
Minimize energy consumption while satisfying comfort requirements
**Goal:**
Minimize energy consumption while satisfying comfort requirements

**Contributions:**
1. A complete reinforcement learning agent that learns and applies a new, adaptive control strategy for a heat-pump thermostat

2. Our agent achieves 7.0%-14.5% yearly energy savings, while maintaining the same comfort level, comparing to a deployed strategy
Simulation Environment

- **GridLAB-D**: A realistic smart-grid simulator, simulates power generation, loads and markets
- **Open-source** software, developed for the **U.S. DOE**, simulates seconds to years
- Realistically models a **residential home**
  - Heat gains and losses, thermal mass, solar radiation and weather effects, uses real weather data recorded by NREL (www.nrel.gov)
Problem Setup

- Simulating a typical *residential* home

- **Goal:** minimize energy consumed by the heat-pump, while satisfying the following comfort spec:

  **Occupants are**
  - 12am-7am: *At home.*
  - 7am-6pm: *Not at home.* (the "*don’t care*" period)
  - 6pm-12am: *At home.*
The Default Thermostat

Temperature (F) vs. Time of Day

- **Upper Bound**: 74°F
- **Lower Bound**: 68°F
The Default Thermostat

The graph shows the temperature over time, with two buffer lines at 76°F and 74°F, an upper bound at 78°F, and a lower bound at 66°F. The temperature fluctuates throughout the day, with a significant rise and fall around 6pm.
The Default Thermostat
The Default Thermostat

Graph showing temperature (F) over time of day, with annotations for buffer, AUX-HEAT, HEAT-PUMP-HEAT, upperBound, and lowerBound.
Can We Just Shut-Down The Thermostat During “don’t-care” Period?

• Effective way to save energy
  – Indoor temp. closer to outdoor → heat dissipation slows down

• Simulating it...

• In this case, the result is:
  – Increased energy consumption
  – Failure to satisfy the comfort spec
Can We Just Shut-Down The Thermostat During “don’t-care” Period?

- Effective way to save energy
  - Indoor temp. closer to outdoor → heat dissipation slows down

- Simulating it...

- In this case, the result is:
  - Increased energy consumption
  - Failure to satisfy the comfort spec

Therefore, people frequently prefer to leave the thermostat on all day.
Can We Just Shut-Down The Thermostat During “don’t-care” Period?

• Effective way to save energy
  – Indoor temp. closer to outdoor → heat dissipation slows down

• Simulating it...

In this case, the result is:
  – Increased energy consumption
  – Failure to satisfy the comfort spec

Therefore, people frequently prefer to leave the thermostat on all day

However, a smarter shut-down should still be able to save energy while maintaining comfort
HEAT PUMP TIPS

- Do not set back the heat pump’s thermostat manually if it causes the electric-resistance heating to come on. This type of heating, which is often used as a backup to the heat pump, is more expensive.
- Install or have a professional install a programmable thermostat with multistage functions suitable for a heat pump.
- Clean or change filters once a month or as needed, and maintain them according to the manufacturer’s instructions.

LONG-TERM SAVINGS TIP

If you heat your home with electricity and live in a moderate climate, consider having a heat pump system to reduce energy requirements. Learn about this technology at energy.gov.

LIMITATIONS FOR HOMES WITH HEAT PUMPS, ELECTRIC RESISTANCE HEATING, STEAM HEAT, AND RADIANT FLOOR HEATING

Programmable thermostats are generally not recommended for heat pumps. In its cooling mode, a heat pump operates like an air conditioner, so turning up the thermostat (either manually or with a programmable thermostat) will save energy and money. But when a heat pump is in its heating mode, setting back its thermostat can cause the unit to operate inefficiently, thereby canceling out any savings achieved by lowering the temperature setting. Maintaining a moderate setting is the most cost-effective practice. Recently, however, some companies have begun selling specially designed programmable thermostats for heat pumps, which make setting back the thermostat cost-effective. These thermostats typically use special algorithms to minimize the use of backup electric resistance heat systems.
Challenges

Desired behavior:
- Maximize shut-down time while staying above the heat-pump slope
- Similarly for cooling (no AUX)

Challenges:
- The heat-pump slope:
  - Is unknown in advance
  - Changes every day
  - Depends on future weather
  - Depends on specific house characteristics
- Action effects are:
  - Drifting rather than constant: since heat is being moved rather than generated, heat output strongly depends on the temperatures indoors, outdoors and along the heat path
  - Noisy due to hidden physical conditions
  - Delayed due to heat capacitors like walls and furniture
- Also, in a realistic deployment:
  - Exploration cannot be too long or too aggressive
  - Customer acceptance will probably depend on worst-case behavior
- Making decisions in continuous, high dimensional space
Our Problem as a Markov Decision Process (MDP)

• States:

• Actions:

• Transition:

• Reward:

• Terminal States:

• Action is taken every **6 minutes**
  – Modeling a realistic lockout of the system
Our Problem as a Markov Decision Process (MDP)

- **States:**
- **Actions:** \{COOL, OFF, HEAT, AUX\}
  \[ 1 : 0 : 2 : 4 \quad \text{consumption (e_a) proportion} \]
- **Transition:**
- **Reward:**
- **Terminal States:**
- Action is taken every 6 minutes
  - Modeling a realistic lockout of the system
Our Problem as a Markov Decision Process (MDP)

- **States:**

- **Actions:** \{COOL, OFF, HEAT, AUX\}
  
  \[
  1 : 0 : 2 : 4 \quad \text{consumption (} e_a \text{) proportion}
  \]

- **Transition:**

- **Reward:**
  
  \[ -e_a - 100000 \Delta^2_{6pm} \quad \text{where:} \]
  
  \[
  \Delta^2_{6pm} := (\text{indoor_temp_at_6pm} - \text{required_indoor_temp_at_6pm})
  \]

- **Terminal States:**

- Action is taken every 6 minutes
  
  – Modeling a realistic lockout of the system
Our Problem as a Markov Decision Process (MDP)

- **States:** ???

- **Actions:** \{COOL, OFF, HEAT, AUX\}
  
  \[ 1 : 0 : 2 : 4 \quad \text{consumption} \ (e_a) \text{ proportion} \]

- **Transition:**

- **Reward:** \(-e_a - 100000 \Delta^2_{6pm}\)  
  where:  
  \[ \Delta^2_{6pm} := (\text{indoor_temp}_{at\_6pm} - \text{required\_indoor\_temp}_{at\_6pm}) \]

- **Terminal States:**

- Action is taken every 6 minutes
  - Modeling a realistic lockout of the system
How Should We Model State?

• Choosing a state representation is an important design decision. A state variable:
  – captures what we need to know about the system at a given moment
  – is the variable around which we construct value function approximations

[Powell 2011]

• Definition 5.4.1 from [Powell 2011]:
  – A state variable is the minimally dimensioned function of history that is necessary and sufficient to compute the decision function, the transition function, and the contribution function.
Our Problem as a Markov Decision Process (MDP)

- **States:** \( <T_{in}, \text{Time}, e_a> \)
- **Actions:** \{COOL, OFF, HEAT, AUX\}
  
  \( 1 : 0 : 2 : 4 \) ← consumption \((e_a)\) proportion

- **Transition:**

- **Reward:** \(-e_a - 100000 \Delta^2_{6pm}\) where:
  
  \( \Delta^2_{6pm} := (\text{indoor_temp}_\text{at}_6\text{pm} - \text{required_indoor_temp}_\text{at}_6\text{pm}) \)

- **Terminal States:**

- Action is taken every 6 minutes
  
  - Modeling a realistic lockout of the system
Our Problem as a Markov Decision Process (MDP)

- **States:** \(<T_{in}, \text{Time}, e_a>\)

- **Actions:** \{COOL, OFF, HEAT, AUX\}

  \[1 : 0 : 2 : 4 \quad \text{consumption} \ (e_a) \proportion\]

- **Transition:**

- **Reward:** \( -e_a - 100000 \Delta^2_{6pm} \) where:

  \[\Delta^2_{6pm} := \text{indoor\_temp\_at\_6pm} - \text{required\_indoor\_temp\_at\_6pm}\]

- **Terminal States:**

- **Action is taken every 6 minutes**

  – Modeling a realistic lockout of the system
Expanding State to Compute the Transition Function

• Can we predict action effects for each of the state variables?

• Current state representation: \(<T_{in}, Time, e_a>\)

• Need to be able to predict \(T_{in}\) and \(e_a\)

• Method: generate simulated data, use cross-validation to test for regression prediction accuracy
Predicting $T_{in}$

- Prediction error is unacceptably high – state $\langle T_{in}, \text{Time}, e_a \rangle$ doesn’t capture enough information
Predicting $T_{in}$

- Prediction error is *unacceptably high* – state $<T_{in}, \text{Time}, e_a>$ doesn’t capture enough information
- Add $T_{out}$ – directly affects $T_{in}$. Prediction error still *unacceptably high*
Predicting $T_{in}$

- Prediction error is **unacceptably high** – state $\langle T_{in}, \text{Time}, e_a \rangle$ doesn’t capture enough information
- Add $T_{out}$ – directly affects $T_{in}$. Prediction error still **unacceptably high**
- Noise explained as hidden home state → add **history** of observable information
  - Previous action

![Graph showing cross-validation error vs. number of features](image)
Predicting $T_{in}$

- Prediction error is unacceptably high – state $<T_{in}, \text{Time}, e_a>$ doesn’t capture enough information
- Add $T_{out}$ – directly affects $T_{in}$. Prediction error still unacceptably high
- Noise explained as hidden home state $\rightarrow$ add history of observable information
  - Previous action
  - Measured $T_{in}$ history of 10 temperatures: $<t_0>$
Predicting $T_{in}$

- Prediction error is *unacceptably high* – state $<T_{in}, \text{Time}, e_a>$ doesn’t capture enough information
- Add $T_{out}$ – directly affects $T_{in}$. Prediction error still *unacceptably high*
- Noise explained as hidden home state $\rightarrow$ add *history* of observable information
  - Previous action
  - Measured $T_{in}$ history of 10 temperatures: $<t_0, t_1>$
Predicting $T_{in}$

- Prediction error is unacceptably high – state $<T_{in}, \text{Time}, e_a>$ doesn’t capture enough information
- Add $T_{out}$ – directly affects $T_{in}$. Prediction error still unacceptably high
- Noise explained as hidden home state → add history of observable information
  - Previous action
  - Measured $T_{in}$ history of 10 temperatures: $<t_0, t_1, t_2>$
Predicting $T_{in}$

- Prediction error is unacceptably high – state $<T_{in}, \text{Time}, e_a>$ doesn’t capture enough information
- Add $T_{out}$ – directly affects $T_{in}$. Prediction error still unacceptably high
- Noise explained as hidden home state → add history of observable information
  - Previous action
  - Measured $T_{in}$ history of 10 temperatures: $<t_0, t_1, t_2, \ldots, t_9>$
Predicting $T_{in}$

- Prediction error is **unacceptably high** – state $<T_{in}, \text{Time}, e_a>$ doesn’t capture enough information
- Add $T_{out}$ – directly affects $T_{in}$. Prediction error still **unacceptably high**
- Noise explained as hidden home state $\Rightarrow$ add **history** of observable information
  - Previous action
  - Measured $T_{in}$ history of 10 temperatures: $<t_0, t_1, t_2, \ldots, t_9>$
  - Resulting state: $<T_{in}, T_{out}, \text{Time}, e_a, \text{prevAction}, t_0, \ldots, t_9>$
Completing the state definition

- Resulting state: \(<T_{in}, T_{out}, Time, e_a, prevAction, t_0, ..., t_9, weatherForecast>\)
- Can we predict the newly added variables?
- Trivially, except for \(T_{out}\)
- Therefore, add \textit{weatherForecast} to state
- \textit{weatherForecast} \textit{doesn’t need to be predicted} in our transition function
- This completes our state definition
- The final resulting state is:
  \(<T_{in}, T_{out}, Time, e_a, prevAction, t_0, ..., t_9, weatherForecast>\)
Our Problem as a Markov Decision Process (MDP)

- **States:** \(<T_{in}, T_{out}, \text{Time}, e_{a}, \text{prevAction}, t_0, ..., t_9, \text{weatherForecast}>\)

- **Actions:** \{COOL, OFF, HEAT, AUX\}

- **Transition:** unknown in advance \(\rightarrow\) learned

- **Reward:** \(-e_{a} - 100000 \Delta^2_{6pm}\) where:
  \[\Delta^2_{6pm} := (\text{indoor_temp_at}_6pm - \text{required_indoor_temp_at}_6pm)\]

- **Terminal States:** \(\{s \mid s.\text{time} = 11:59pm\}\)

- **Action taken every 6 minutes**
  - Modeling a realistic lockout of the system
- **State space is continuous and high dimensional**
Agent Operation

First 3 days: exploration

Choose Random Action

Record Action Effect: <s,a,s'>

Observe Resulting State

Starting day 4: energy-saving setback policy

Choose Best Action (TreeSearch)

If Midnight? Update House Model From data (regression)

Record Action Effect: <s,a,s'>

Observe Resulting State

First 3 days: exploration

Choose Random Action

Record Action Effect: <s,a,s'>

Observe Resulting State

Starting day 4: energy-saving setback policy

Choose Best Action (TreeSearch)

If Midnight? Update House Model From data (regression)

Record Action Effect: <s,a,s'>

Observe Resulting State
Agent Operation

First 3 days: exploration

Choose Random Action

Record Action Effect: \( <s,a,s'> \)

Observe Resulting State

Starting day 4: energy-saving setback policy

Choose Best Action (TreeSearch)

If Midnight? Update House Model From data (regression)

Record Action Effect: \( <s,a,s'> \)

Observe Resulting State

First 3 days: exploration

Choose Random Action

Record Action Effect: \( <s,a,s'> \)

Observe Resulting State

Starting day 4: energy-saving setback policy

Choose Best Action (TreeSearch)

If Midnight? Update House Model From data (regression)

Record Action Effect: \( <s,a,s'> \)

Observe Resulting State
Exploration

- Random actions for 3 days
- Could use more advanced exploration policy
- However, this is still a realistic setup
Exploration

• Random actions for 3 days

• Could use more advanced exploration policy

• However, this is still a realistic setup
  – For instance when occupants are traveling during the weekend
Agent Operation

First 3 days: exploration

Choose Random Action

Record Action Effect: \( <s, a, s'> \)

Observe Resulting State

Starting day 4: energy-saving setback policy

Choose Best Action (TreeSearch)

If Midnight? Update House Model From data (regression)

Record Action Effect: \( <s, a, s'> \)

Observe Resulting State
Update House Model from Data

• Every midnight, use all the recorded data $\langle s, a, s' \rangle$ to estimate the house’s transition function.

• Linear Regression to estimate $\langle s, a \rangle \rightarrow s'$.
Agent Operation

First 3 days: exploration

Choose Random Action

Record Action Effect: \(<s,a,s'>\)

Observe Resulting State

Starting day 4: energy-saving setback policy

Choose Best Action (TreeSearch)

If Midnight? Update House Model From data (regression)

Record Action Effect: \(<s,a,s'>\)

Observe Resulting State
Choosing the Best Action

- Dealing with continuous high-dimensional state space
- Impractical to compute a value function
- Run a tree search at every step
- Choose the first action of the best search as the next action
Safety Buffer in a Tree Search

$C \sim 0$

$C \sim 2\sigma$
Results

• Simulate 1 year under different weather conditions
• 21 residential homes of sizes 1000-4000 ft$^2$
• Using real weather data recorded in NYC, Boston, Chicago

• Why cold cities? Since heating consumes 2x-4x more energy
Temperature Graphs – Learned Setback Policy
Energy Savings

Excluding Exploration Days

Excluding Exploration Days

Energy savings (kWh)

Energy savings (%)

House Size (sq Ft.)

House Size (sq Ft.)

NYC
Boston
Chicago
Comfort Performance

• In more than 22,000 simulated days
Related Work

- [Rogers et al. 2011] – adaptive thermostat that tries to minimize price & peak demand rather than the total amount of energy.

- [Hafner and Riedmiller 2011; Kretchmar 2000] – use RL to tune an HVAC system.

- [T. Peffer et al. 2011] – How people use thermostats in homes

- Learning thermostats in commercial companies
  - NEST, Honeywell...
  - Technical details and actual performance are not published
Summary

- A complete, adaptive, RL agent for controlling a heat-pump thermostat
- Techniques:
  - Carefully defined the problem as an MDP
  - Carefully chose a state representation
  - Using an efficient, specialized tree-search
- Experiments run on a range of homes and weather conditions
- Achieves 7%-14.5% yearly energy savings in simulation, while satisfying comfort requirements, comparing to the deployed strategy

Thank you!
BACKUP
Ablation Analysis

Table 1: Ablation Analysis

<table>
<thead>
<tr>
<th>Removed Feature</th>
<th>Analysis Type</th>
<th>Energy Consumption (kWh)</th>
<th>Comfort Violations (#)</th>
<th>Range of 6pm Temp.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>prevAct+hist+ conf</td>
<td>1112(+9.5%)</td>
<td>232</td>
<td>60.1-84.4</td>
</tr>
<tr>
<td></td>
<td>prevAct+hist</td>
<td>1070(+5.4%)</td>
<td>193</td>
<td>60.8-80.9</td>
</tr>
<tr>
<td></td>
<td>conf</td>
<td>1024(+0.8%)</td>
<td>138</td>
<td>67.5-78.3</td>
</tr>
<tr>
<td></td>
<td>hist</td>
<td>1016(+0.0%)</td>
<td>133</td>
<td>67.1-77.7</td>
</tr>
<tr>
<td></td>
<td>prevAct</td>
<td>1015(+0.0%)</td>
<td>65</td>
<td>67.8-76.5</td>
</tr>
<tr>
<td>Other conf. bounds</td>
<td>2σ</td>
<td>1090(+7.3%)</td>
<td>29</td>
<td>69.0-78.5</td>
</tr>
<tr>
<td></td>
<td>c = 2</td>
<td>1039(+2.3%)</td>
<td>27</td>
<td>69.0-77.8</td>
</tr>
<tr>
<td>Final Agent</td>
<td></td>
<td>1015</td>
<td>23</td>
<td>68.8-76.6</td>
</tr>
</tbody>
</table>

- Removing features and their combinations
  - State features:
    - prevAct: previousAction
    - Hist: temperature history $t_0, \ldots, t_9$
  - conf: confidence buffer

- Setting other values to the confidence bound