2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Learning-based Model Predictive Control and User Feedback in Home Automation

Christopher C. W. Ham, Surya P. N. Singh, and Michael Kearney

Abstract— **Air conditioning systems are generally the largest systems in a home, both physically and energetically, and thus central to home automation efforts. Finding a balance between user comfort and efficiency is a complex problem given the considerable variation present. This paper focuses on comfort, as opposed to absolute temperature, as gauged using simple, sparse user inputs. A thermal model of the house is learned which accounts for weather data and exogenous factors such as occupancy. By incorporating user feedback, a Learning-Based Model Predictive Controller (LBMPC) is able adapt to home conditions and more efficiently operate the system. In contrast to previous efforts which operate in office spaces and to a set point, this work is adapted and tested in a typical home environment and closes a control loop on user comfort. The controller considers that the user's comfort levels may change during the day, for example when the user is in bed, or not at home. It shows that complex systems may be automated without extensive tweaking by the user and in a manner that considers user comfort, time of day, and related factors to reduce energy consumption.**

I. INTRODUCTION

Despite considerable advances in home automation hardware, the application of these systems remains challenging and limited due to the extent of user variation. In other words, assumptions that may work for some people may not necessarily work for others. Trade-offs between comfort and energy consumption often involve these assumptions, hence the need for direct user feedback, preferably in a simple, convenient manner.

In home automation there is a strong motivation to increase the efficiency of HVAC (heating, ventilation, and air conditioning) systems. In Australian homes, for example, heating and cooling is the largest consumer of energy [1, 2]. Even though they are not efficient, thermostats remain the popular means to regulate temperature, in part because encoding user preferences and assumptions are difficult to automate.

Consider work attention has been focused on automating these systems to maximize energy efficiency, particularly of the split-system air conditioners that are typically found in Australian homes [3]. Aswani et. al [4] focused on an optimal control scheme using Learning-Based Model Predictive Control (LBMPC) that makes use of weather predictions. However, this work makes fixed assumptions about temperature ranges for user comfort. Erickson and Cerpa [5] improve the performance of this controller by predicting future occupancy levels, but their proposed system would be expensive to set up in a residence because of the extra parts, labor and expertise required.

Traditionally, HVAC control systems focus on regulating a temperature to a set point when, in fact, it is the users' comfort levels that are of interest. Towards this, the Nest thermostat [6] attempts to learn the habits of its users and estimates what the set point should be at any time. However, it still relies on the user working out what temperature they prefer at what times.

This paper introduces a method that closes this loop by acting on user feedback rather than the temperature. For example, 25°C can feel hot in winter yet cool in summer; or someone at home is less affected by lower temperatures when they're in bed.

In HVAC systems assumptions are usually made about the user's comfort levels; some attempt to keep the temperature between 22°C and 24°C, for example. By relaxing these assumptions, it is possible to reduce energy consumption while improving comfort. Many studies have shown increased efficiency for tightly controlled temperatures [4, 7, 8]. The work in this paper, however, shifts the focus from tightly controlled temperatures to user comfort. By receiving occasional feedback from the user, the controller can learn their range of comfort. The control can then be relaxed to reduce energy consumption while still ensuring the user satisfaction.

The controller also makes considerations for the user's comfort at different times of day. Homes also differ from offices in that the requirements of the users are less consistent. By learning these daily patterns the controller tightens the constraints when appropriate. For example, if the user is not at home then it does not matter what the temperature is; if the user is in bed, the room can be cooler than usual and still maintain their comfort.

The work of Aswani et. al [4] showed the effectiveness of LBMPC in a computer lab at Berkeley. It is used as a starting point and is extended by taking advantage of the personalisation that's possible in home automation.

The system has been designed to be affordable and easy to set up with little technical knowledge. It does not require prior knowledge of the space into which it is being deployed. During setup, the goal is to learn the parameters following model which predicts the future temperature based on the current state of the system:

C. Ham and S. Singh are in the School of Information Technology and Electrical Engineering and M. Kearney is in the School of Mechanical and Mining Engineering at The University of Queensland, Brisbane QLD 4072. This research was performed, in part, at the Robotics Design Laboratory. Correspondence should be addressed to christopher.ham@uqconnect.edu.au

$$T[n+1] = k_T T[n] + k_w w[n] + k_u u[n] + q[n] + \varepsilon[n] \quad (1)$$

Where $T[n]$ is the temperature at time, n; $w[n]$ is the outside temperature at time, n; $u[n]$ is the duty cycle of the air conditioner at time, n; $q[n]$ is the nonlinear occupancy level at time n; $\varepsilon[n]$ is the error in the model.

Here, k_T is the time constant for the temperature of the room, k_w is the time constant for the heat transfer between the room and the outside, k_u is the change in temperature over 15 minutes (°C) proportional to the duty cycle of heating or cooling input to the system, and $q[n]$ is the change in temperature in 15 minutes due to unmodelled factors like people, open windows, and computers [4].

The parameters, k_T, k_W, and k_u are learned by observing the response of the system when given a random sequence of duty cycle inputs for a day.

In this work, LBMPC is extended by encouraging the user to provide simple feedback ("Too hot" or "Too cold") via a smartphone app (Fig. 1) allowing the controller to estimate a user's range of comfort by testing for these boundaries. Fig. 2 shows how the estimated comfort bounds of user change with feedback and time. The first feedback from the user initially pulls down the maximum temperature for 11am to 24°C, but is relaxed the following day to 26°C. A second input from the user at this time of day holds the comfort bounds at just over 24°C in the days following.

In this paper "Comfort bounds" and "constraints" are interchangeable. Particularly, "constraints" is used in the context of optimisation.

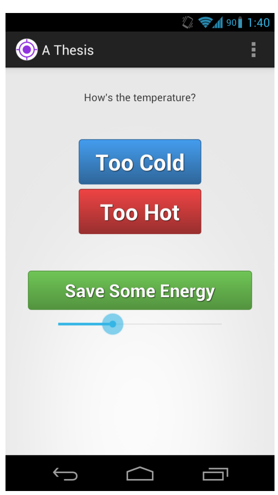

Fig. 1. A screenshot of the smartphone app used in this paper

In Section II, the problem is formulated and semiparametric regression, LBMPC, extensions to LBMPC, and algorithms are described. In Section III, experiments show how the comfort bounds are estimated from user feedback, and how the proposed controller performs in a real home environment. Section IV summarises the extensions to LBMPC and the methods used to close the loop on user comfort. Additionally future work is discussed.

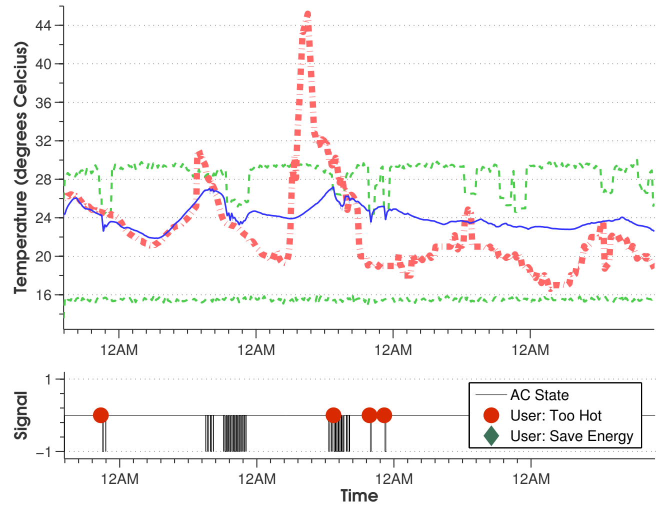

Fig. 2. Illustrating the controller testing the user's comfort bounds. Solid, blue line: inside temperature. Thick, dashed, red line: outside temperature. Thin, dashed, green line: comfort bounds.

II. METHODS

A. Learning-Based Model Predictive Control

Model predictive control is an advanced control technique that is commonly to control systems with constraints to minimize a given objective function. This framework maps directly to this problem, where the object to minimize is the energy used by the HVAC system and the user's comfort form the constraints [9].

The learning-based variant of MPC (denoted LBMPC), proposed by Aswani et. al [4, 10], is selected for its ability to handle disturbances and nonlinear systems. It uses an empirical model of the system can be obtained by using semiparametric regression [10]; and no prior knowledge of the system is required to fit it to the basic thermal model described in Eq. (1).

Once a model of the system has been obtained, the future inside temperature can be predicted given a weather forecast of the outside temperature and sequence of HVAC inputs. Using this, the cost function Eq. (2) is minimised by selecting the optimum inputs.

LBMPC operates at each control cycle by:

- measuring the system,
- learning the current occupancy by noting the residual of the expected and current temperatures,
- updating the model using a regression approach that will be described in the sequel,
- finding the sequence of inputs that minimises energy consumption and user discomfort,
- applying the first input of this sequence and discarding the rest.

For the cost function, the learned occupancy is used to obtain a tighter estimation of the system ($\tilde{T}$). The constraint function uses the mean of the occupancy ($\bar{T}$) that was determined while learning the model of the system. Using the mean is safer and more stable when considering constraints [4, 10]. The basic optimisation problem is given by,

$$\min u[\cdot] \sum_{k=0}^{N} p \cdot (\tilde{T}[m+k] - T_d)^2 + (r+\lambda) \cdot u[m+k] \quad (2)$$

$$\text{s.t. } \tilde{T}[m+i] = k_T \cdot \tilde{T}[m+i-1] + k_u \cdot u[m+i-1] + k_w \cdot w[m+i-1] + \hat{q}[m+i-1] \quad (3)$$

$$\bar{T}[m+i] = k_T \cdot \bar{T}[m+i-1] + k_u \cdot u[m+i-1] + k_w \cdot w[m+i-1] + q_{mean} \quad (4)$$

$$\bar{T}[m+i] \in [T_{min}[m+i], T_{max}[m+i]] \quad (5)$$

$$u[m+i-1] \in [0, u_{maxDuty}] \quad (6)$$

Where m is the current time, N is the control horizon (chosen to be 20; or 5 hours), p is the weighting on the temperature error and T_d is the desired temperature.

B. Semiparametric Regression

Semiparametric regression is used to estimate the coefficients of a model based on input and output data. This approach allows unmodelled factors such as occupancy, open windows, and computers to be captured by the model, thereby reducing the error between the predicted and real values.

The coefficients from the thermal model discussed earlier (Eq. (1)) can be determined using the following method. The process described here is an extension of [11].

The conditional expectations are defined as:

$$\hat{T}[n] = \mathbb{E}[T[n]|n] \quad (7)$$

$$\hat{w}[n] = \mathbb{E}[w[n]|n] \quad (8)$$

$$\hat{u}[n] = \mathbb{E}[u[n]|n] \quad (9)$$

By substituting the expected values into Eq. (1) the following equation is obtained:

$$\hat{T}[n+1] = k_T\hat{T}[n] + k_w\hat{w}[n] + k_u\hat{u}[n] + \mathbb{E}[q[n]|n] \quad (10)$$

Since $q[n]$ is highly nonlinear with respect to time,

$$\mathbb{E}[q[n]|n] = q[n],$$

and since ε is zero mean,

$$\mathbb{E}[\varepsilon[n]|n] = 0.$$

So by subtracting Eq. (10) from Eq. (1), this nonlinear term can be removed while estimating the coefficients:

$$T[n+1] - \hat{T}[n+1] = k_T(T[n] - \hat{T}[n]) + k_w(w[n] - \hat{w}[n]) + k_u(u[n] - \hat{u}[n]) + \varepsilon[n] \quad (11)$$

Now, the coefficients can be computed by minimising the error, ε:

$$(\hat{k}_T, \hat{k}_w, \hat{k}_u) = \arg\min L(k_T, k_w, k_u) \quad (12)$$

where:

$$\begin{aligned} L(k_T, k_w, k_u) &= ||\varepsilon(k_T, k_w, k_u)||_2 \\ &= ||T[n+1] - \hat{T}[n+1] - k_T(T[n] - \hat{T}[n]) \\ &\quad - k_w(w[n] - \hat{w}[n]) - k_u(u[n] - \hat{u}[n])||_2. \end{aligned} \quad (13)$$

After obtaining the coefficients of the model the occupancy term can be estimated by

$$\hat{q}_n = \hat{T}[n+1] - k_T\hat{T}[n] - k_u\hat{u}[n] - k_w\hat{w}[n] \quad (14)$$

since $\hat{q}[n]$ should be the discrepancy between the predicted value and the actual value.

The original model variables are index by time, and so obtaining the expected values of these variables, $\hat{T}$, $\hat{w}$, and $\hat{u}$ are equivalent to kernel smoothing over time [11]. Similar to the Berkeley paper, these values are estimated using the Nadaraya-Watson estimator.

The Nadaraya-Watson estimator was used to statistically smooth the input data. An appropriate bandwidth, h, was determined for the kernel smoothing regression using the guidelines provided in [12];

$$h = 1.06\sigma N^{-1/5}, \quad (15)$$

where σ is the standard distribution of the entire data set.

C. Approach

The input that the controller passes to the air conditioner is a pulse-width modulated (PWM) signal with a period of 15 minutes. This period was chosen so as not to over-cycle the unit. Additionally, the duty cycle is limited to 70% in order to respect the limits of the air conditioner and avoid damage.

The learned temperature bounds are reassessed for different times of the day. The idea is that the absolute set-point of a room is not critical, and that when and where the set-point should be placed is highly subjective. By this philosophy, the user is never shown the temperature or asked to input a desired temperature. Instead they simply inform the system when they are too hot or too cold.

The work in this paper differs from the work at Berkeley [4] in other ways. The control was complicated by the local climate and the fact that the occupancy factor of a room at home is much less than that of a computer lab. In Brisbane the temperature can go from 16 to 28°C in a 12 hour period. This often means that the control has to handle both modes of cooling and heating in the same control horizon. This climate is difficult to handle because the model of the system changes depending on the mode of input. Additionally computer labs often have 20 or more computers and bodies forcing the room temperature higher; homes, in contrast, have a less dense heating load than labs [13]. This difference makes the control of homes more multi-modal because they have both heating and cooling requirements that need to be considered in the same control horizon.

The process of estimating the coefficients is now described. This provides an extension to the approach presented in [4].

1) Dual-Mode Model Regression: Due to the climate of South-East Queensland a combination of heating and cooling is required to keep the inside temperature within the defined bounds - even in the same control horizon. As a result, the semiparametric regression for a dual-mode input model requires an important modification to the method used for the single mode input model.

The modification splits the k_u constant into separate constants for cooling and heating (k_c and k_h). The input vector is now allowed to vary from -70% to 70% - instead of 0% to 70%. The sign of the input is used to determine the mode of forcing and thus which constant to use. A positive duty cycle is for heating and a negative duty cycle is for cooling. Note that the magnitude if the input is used for the model in the algorithms and so the constants k_c and k_h are signed. The main reason for doing it this way is so that the effect of each constant is more obvious. The algorithm is formalised by Algorithm 2.

Algorithm 1 Determining which k_u to use

```
function GETKU(u)
    u ← the current input value
    k_u ← the constant to use
    if u < 0 then
        k_u ← k_c
    else
        k_u ← k_h
    end if
return k_u
end function
```

Algorithm 2 Optimisation Function to Determine Dual Mode Model

$k_u \leftarrow getKu(u[n])$
$\hat{k}_u \leftarrow getKu(\hat{u}[n])$
return $\Big\| T[n+1] - \hat{T}[n+1] - k_T(T[n] - \hat{T}[n]) \ldots$
$-k_w(w[n] - \hat{w}[n]) - k_u\Big|u[n]\Big| + \hat{k}_u\Big|\hat{u}[n]\Big|\Big\|_2$

After obtaining the coefficients for the dual-mode model the estimated occupancy is calculated in a similar fashion to a single-mode model [4] except the value of k_u used depends on the sign of the input. This is shown in Algorithm 3.

Algorithm 3 Calculating $\hat{q}$ for Dual Mode Models

$k_u \leftarrow getKu(u[n])$
$\hat{q}_n = \hat{T}[n+1] - k_T\hat{T}[n] - k_u\,|\hat{u}[n]| - k_w\hat{w}[n]$ ▷ Note the absolute value
return $\hat{q}_n$

2) Dual-mode Model Control: As before, the control considers the two modes of forcing (heating and cooling) by allowing negative input values in the optimisation step. The sign of the input is used to select the appropriate model (k_u constant). To predict future temperatures for the optimiser, Algorithm 4 is used.

Algorithm 4 Predicting the Temperature for a Dual Mode Model

```
function GETPREDICTEDTEMP(T_0, u, w, q)
    N ← time steps in control horizon
    u ← inputs for current horizon
    w ← forecast of outside temperatures
    k_w ← model coefficient for outside temp.
    k_u ← model coefficient for input
    q ← the occupancy
    T_0 ← current temperature of system
    T ← predicted temperatures for the horizon

    T[0] ← T_0
    for i = 1 to N − 1 do
        k_u ← getKu(u[i − 1])
        T[i] = k_T·T[i−1] + k_w·w[i−1] + k_u·|u[i − 1]| + q
    end for
return T
end function
```

3) Learning the User's Comfort Range: In Aswani's implementation [4], the constraints are set to keep the temperature bound between 20 and 24°C. These temperatures are the bounds of what is considered to be a comfortable range for most people [14]. By learning the comfort range of the user these assumptions can be removed, leading to potential energy savings.

When learning the user's comfort bounds, the desired effect is for the system to "test" these bounds so as to encourage the user's input. When the user informs the system they are too hot or too cold, the controller should try ensure the user's immediate comfort. To do this, the learned comfort bounds are relaxed each day until the user has provided sufficient feedback. The pseudocode to achieve this is described in Algorithm 5.

4) Relaxing the Constraints During Transition: When the controller has just been switched on or when the comfort bounds have been recently changed due to user input, the system will attempt to transition into a state that satisfies the comfort bounds. If the inside temperature is initially outside of these bounds, the optimiser would ordinarily fail - being unable to satisfy the constraints.

The constraints on the first n steps are removed. To begin with $n = 0$, then each time the optimiser fails by violation of the constraints, n is incremented and the optimisation is run again. The algorithm is detailed in Algorithm 6.

III. EXPERIMENTS

The components and interactions of the system used in the experiments is illustrated in Fig. 3. Note that emphasis is placed on making the system inexpensive to install in existing homes. Such a requirement necessitates that the system be installable without professional help.

Algorithm 5 Learning a User's Comfort Range

function FINDBOUNDS(T, t, B)
  $T \leftarrow$ feedback temperatures for a boundary
  $t \leftarrow$ times the feedback was recorded
  $d \leftarrow$ number of days since each time in t
  $B \leftarrow$ the boundary these data points are for

  $N_{min} \leftarrow$ min number of data points required
  $D_{max} \leftarrow$ number of days to weight the data
  $W_{max} \leftarrow$ weight given to most recent data
  $T_{new} \leftarrow$ new comfort limit for this boundary
  $P_{dist} \leftarrow$ fitted normal distribution parameters

  $d \leftarrow$ daysSince(t)

  if $B =$ High **then**
    $T_{inject} \leftarrow 27°\text{C}$
  else ▷ Low
    $T_{inject} \leftarrow 15°\text{C}$
  end if

  $W \leftarrow \lceil W_{max} \times \frac{D_{max}-d}{D_{max}} \rceil$ ▷ Weights
  $W \leftarrow \max(W, 1)$
  $T \leftarrow$ duplicate(T, W) ▷ based on W

  while length(T) $< N_{min}$ **do**
    $T \leftarrow$ cat($T, T_{inject} + noise$) ▷ Noise stops a singularity from breaking the normal-fitting function
  end while

  $P_{dist} \leftarrow$ fitdist(T, 'normal')
  if $B =$ High **then**
    $T_{new} \leftarrow$ cdf(P_{dist}) < 0.10
  else
    $T_{new} \leftarrow$ cdf(P_{dist}) > 0.90
  end if
return T_{new}, P_{dist}
end function

Algorithm 6 Relaxing Constraints During Transitions

$m \leftarrow$ the mask vector
$u_0 \leftarrow$ initial input vector
$u \leftarrow$ optimal input
$r \leftarrow$ minimisation error
$f \leftarrow$ minimisation flag
$C_{max} \leftarrow$ maximum constraints to mask

for $k = 0$ to C_{max} **do**
  $m(1:k) \leftarrow true$
  $(u, r, f) =$ argmin(u_0) ▷ Run the minimisation. Uses m internally
  if ($f =$ success) **then**
    break
  end if
end for

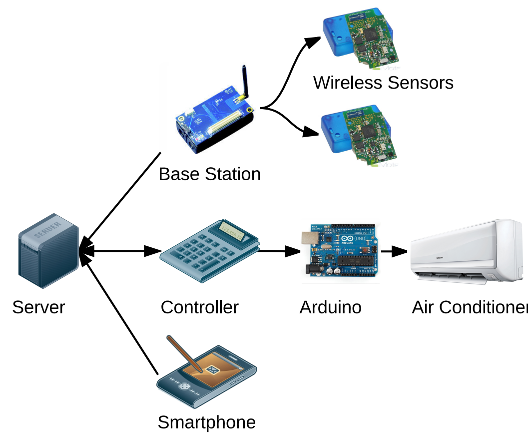

Fig. 3. Interactions between the server and other components.

A laptop is used to run the controller and server, it communicates with the Arduino via USB serial. The Arduino and a simple infrared control circuit are used to learn and emit the appropriate commands to control the air conditioner. Wireless sensors are used to make measurements of the inside and outside temperatures.

The server essentially supports the entire setup. All important information and events are recorded in a log file on the server and can be remotely read by any component of the system. This data includes the temperature measurements, AC duty cycles, user input from the smartphone app and calculated user temperature bounds.

A. Dual-Mode Model

Fig. 4 shows the state of the system while random input was being applied. By applying semiparametric regression to the results, a model of the system can be obtained. This can be done without the need for any user interaction or professional. The coefficients of the model obtained for this system are shown in Table I.

TABLE I
MODEL COEFFICIENTS FOR DUAL MODE INPUT

k_T	k_c	k_h	k_w	$\bar{q}$
0.9161	-1.5964	1.4642	0.0258	1.4865

Using this learned dual-mode model, Fig. 5 shows the system successfully keeping the temperature inside the generalised comfortable range. The desired temperature was 22°C. It shows that the controller is able to keep the temperature inside the constraints by considering both cooling and heating.

B. Learning the Range of User Comfort

As described in Section II-C, the controller can be configured to continue relaxing the comfort bounds until a certain number of feedback data points are received. In this experiment the controller only requires two feedback data points at a certain time of the day. This number of required data points is chosen so that the user is not annoyed by regularly becoming uncomfortable.

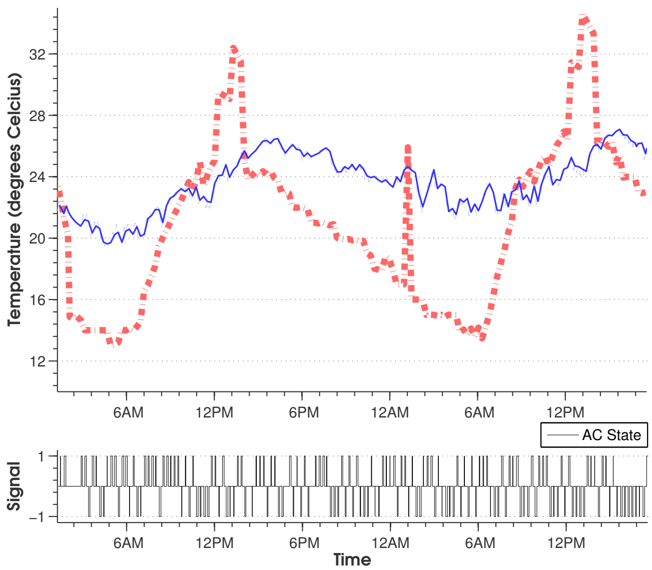

Fig. 4. A signal of random duty cycle [-70, 70]%) forces the system in order to determine the model. Solid, blue line: inside temperature. Thick, dashed, red line: outside temperature.

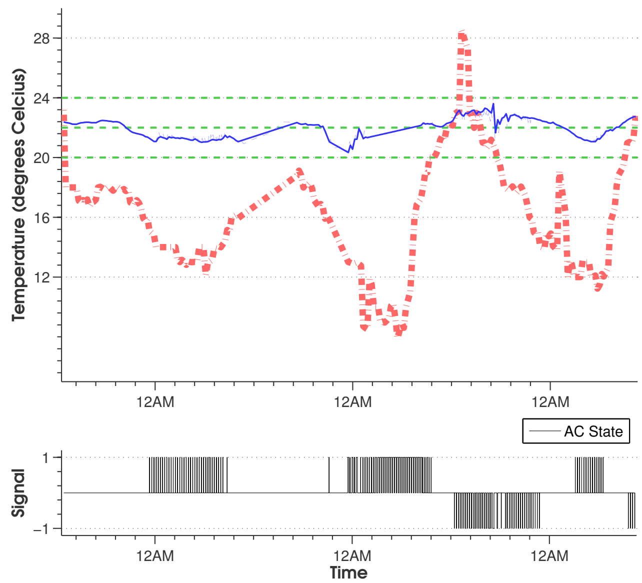

Fig. 5. The output of the system under dual mode control. The optimiser is able to mix different control modes in the same control horizon. Solid, blue line: inside temperature. Thick, dashed, red line: outside temperature. Thin, dashed, green line: comfort bounds. The set point is 22°C.

Figure 2 shows the user expressing discomfort. The comfort bounds at the same time the next day are slightly relaxed. The user expresses discomfort again. After which the comfort bounds remain the same in the following days.

C. Constraint Relaxation During Transition

Figure 6 shows the system responding to sudden changes in the temperature constraints due to user input (more information in Section III-B). The system tries to drive the temperature to inside these constraints as quickly as possible. If this relaxation is not applied, the optimiser is not able to find a solution and no actuation at all would occur.

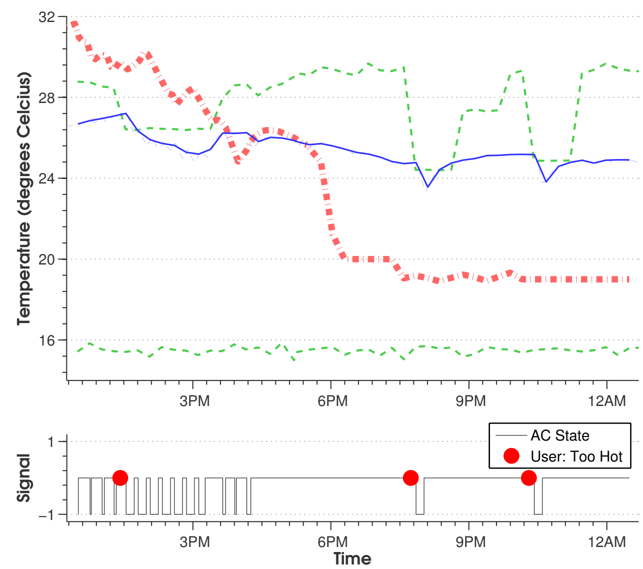

Fig. 6. When feedback is received, the newly learned constraints will go out of bounds of the current temperature. The controller cannot immediately satisfy this, and so relaxations are allowed in the next control cycle(s). Solid, blue line: inside temperature. Thick, dashed, red line: outside temperature. Thin, dashed, green line: comfort bounds.

IV. CONCLUSIONS

A smartphone application coupled with an adaptive control approach based on semiparametric regression has been shown to estimate user comfort level at different times in the day without explicitly asking for an exact temperature set point and operation time. Upon receiving limited feedback from the user ("Too Hot" or "Too Cold") the controller attempts to minimize discomfort by immediately adjusting the constraints. If the certainty of these comfort bounds is low (measured by the number of feedback inputs received by the user), they are relaxed over several days. In doing so, further feedback is encouraged and so the comfort bounds are relaxed less, until no further feedback is required.

Learning-based model predictive control (LBMPC) is an effective control algorithm to handle constraint trajectories. It is able to calculate future constraints and predict an optimum input over a control horizon to satisfy those constraints.

LBMPC can be successfully adapted for a home environment by modifying the modelling and optimisation steps to account for two modes of air conditioning - heating and cooling. This is accommodated for by considering both negative values (cooling) and positive values (heating).

The controller is further improved by allowing relaxations in the control when the comfort bounds are exceeded during transitions - when the constraints are changed or the system is initially switched on. Once the system transitions into a steady control state this is no longer required.

An obvious extension to this work will be to allow the controller to consider the comfort of multiple users. Integration of the smartphone application would allow the system to deduce which users are home - there is no need to satisfy the comfort of anyone away from home.

Other future work will investigate an optimisation approach to weigh comfort against energy usage. The advantage of such an approach is that the optimiser's emphasis on comfort or energy saving can be easily modified by providing the user with another feedback option on the smartphone app.

In conclusion, it is shown that it is possible to close the control loop on the user's comfort using a smartphone application, and that LBMPC is able to regulate to this while considering energy consumption.

References

[1] E. Strategies, L. Harrington, R. Foster, and G. Wilkenfeld, *Energy use in the Australian residential sector 1986-2020*. Department of the Environment, Water, Heritage and the Arts, 2008, pp. 24–25. [Online]. Available: goo.gl/xMGhj.

[2] G. of South Australia. (2011). How energy is used in the home, [Online]. Available: goo.gl/HDkQc.

[3] C. Chiou, C. Chiou, C. Chu, and S. Lin, "The application of fuzzy control on energy saving for multi-unit room air-conditioners," *Applied Thermal Engineering*, vol. 29, no. 2, pp. 310–316, 2009.

[4] A. Aswani, N. Master, J. Taneja, D. Culler, and C. Tomlin, "Reducing transient and steady state electricity consumption in hvac using learning-based model-predictive control," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 240–253, 2012.

[5] V. Erickson and A. Cerpa, "Occupancy based demand response hvac control strategy," in *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*, ACM, 2010, pp. 7–12.

[6] Y. Matsuoka. (2012). Teaching nest to save energy, [Online]. Available: http://www.nest.com/blog/2011/11/08/teaching-nest-to-save-energy/.

[7] F. Oldewurtel, D. Gyalistras, M. Gwerder, C. Jones, A. Parisio, V. Stauch, B. Lehmann, and M. Morari, "Increasing energy efficiency in building climate control using weather forecasts and model predictive control," in *10th REHVA World Congress Clima*, 2010, pp. 9–12.

[8] W. Burke and D. Auslander, "Low-frequency pulse width modulation design for hvac compressors," in *The ASME*, 2009.

[9] J. M. Maciejowski, "Predictive control with constraints," 1999.

[10] A. Aswani, H. Gonzalez, S. Sastry, and C. Tomlin, "Provably safe and robust learning-based model predictive control," *arXiv preprint arXiv:1107.2487*, 2011.

[11] A. Aswani, N. Master, J. Taneja, V. Smith, A. Krioukov, D. Culler, and C. Tomlin, "Identifying models of hvac systems using semiparametric regression," in *Proceedings of the American Control Conference*, 2012.

[12] R. Gutierrez-Osuna, *L7: kernel density estimation*, Texas A+M University, Texas, USA, 2012. [Online]. Available: http://research.cs.tamu.edu/prism/lectures/pr/pr_l7.pdf.

[13] B. Olesen and G. Brager, "A better way to predict comfort: the new ashrae standard 55-2004," 2004.

[14] H. Burroughs and S. Hansen, *Managing Indoor Air Quality*. Fairmont Press, 2011, pp. 148–150, ISBN: 9780881736618. [Online]. Available: http://books.google.com.au/books?id=KBbHmp2yisEC.