Optimal Tracking Strategies for Uncertain Ensembles of Thermostatically Controlled Loads

Sribalaji Coimbatore Anand

Master of Science Thesis

Delft University of Technology

Delft Center for Systems and Control

Optimal Tracking Strategies for Uncertain Ensembles of Thermostatically Controlled Loads

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

Sribalaji Coimbatore Anand

July 4, 2019

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Delft University of Technology
Department of
Delft Center for Systems and Control (dcsc)

The undersigned hereby certify that they have read and recommend to the Faculty of Mechanical, Maritime and Materials Engineering (3mE) for acceptance a thesis entitled

Optimal Tracking Strategies for Uncertain Ensembles of Thermostatically Controlled Loads

by

Sribalaji Coimbatore Anand

in partial fulfillment of the requirements for the degree of

Master of Science Systems and Control

Dated: July 4, 2019

Supervisor(s): ________________________
dr. Simone Baldi

Reader(s): ________________________
dr. Sergio Grammatico

____
dr. Peyman Mohajerin Esfahani

Abstract

Traditional centralized power plants have limited ability to adapt to the varying power demands caused due to the increasing deployment of renewable energy sources. For power grids, willing to increase the use of renewable energy and thereby decrease the energy bills, demand side energy management could act as an effective solution. Demand side energy management of the power grid refers to the process of regulating the power demands of the devices it serves. A large fraction of this power demand on the grid lines is caused due to Thermostatically Controlled Loads (TCL) such as residential refrigerators, electric water heaters, air conditioners, industrial heaters, ovens, etc. Traditionally, the energy management of these devices is achieved using model predictive control and linear quadratic regulation. To better handle the system heterogeneity and computation costs, model-free adaptive control algorithms are explored.

A homogeneous population of TCL, modeled as a second-order system is considered for the study. Its power tracking capabilities, using both state and output feedback control, are discussed along with its limitations. Power tracking is achieved optimally by varying its temperature set-point. A more rational system representation for a TCL population based on state bins is adopted. Similar power tracking capabilities are studied using a non-linear control approach. It is concluded that adaptive optimal control strategies can be effectively used to regulate the power demands of TCL populations. Numerical simulations are provided for varying set-points, input weight, and parameter heterogeneity. Finally, a policy iteration algorithm based on output feedback is proposed with stability analysis.

Contents

List of Figures

List of Tables

Acknowledgements

This research thesis would not have been possible without many incredible supports that I have had in my life. I would like to appreciate the efforts that they have invested for me by thanking them here. I would like to thank

Dr. Ir. Simone Baldi for his incredible support, encouragement, and patience. I admit that working on such a thesis with his guidance made me understand *systems and control* with a greater level of appreciation and intuition.

My parents Mrs. S. Chithra and Mr. C. R. Anand for their encouragement in supporting me to pursue my dreams in spite of many odds.

The Delft Centre of Systems and Control (DCSC) for providing me with an amazing learning platform.

Dr. Aparna Lashmanan - my role model, for supporting me in many ways and by providing me with many valuable life lessons.

My supervisors before joining TU Delft who had played significant roles in my life imparting moral, ethical and research values.

- Dr. K. Madhava Krishna, IIIT Hyderabad, India.
- Mr. Jim Seelan, Tata Consultancy services, India.
- Dr. S. Rakesh Kumar, SASTRA University, Thanjavur, India.
- Dr. T. Asokan, IIT Madras, India
- Dr. Anjan Kumar Dash, SASTRA University, Thanjavur, India.

My friends who has been by my side providing honest feedback and helping me at times when necessary.

Harisubramanyabalaji . S. P
Nguyen Hai Anh
Mohamed Abdelmoumni
Ben Zwerink Arbones
Saravanan Nagesh
Prithvi. L. T.

Delft, University of Technology
July 4, 2019

Sribalaji Coimbatore Anand

Notation

Throughout this technical report,

$\mathbb{R}_+$ and $\mathbb{Z}_+$ denote the sets of non-negative real numbers and non-negative integers, respectively.

Vertical bars $|.|$ represent the Euclidean norm for vectors, or the induced matrix norm for matrices.

For any piecewise continuous function u, $||u||$ denotes $\sup|u(t)|, \forall t \geq 0$.

$\otimes$ indicate Kronecker product.

$vec(A)$ is defined to be the mn-vector formed by stacking the columns of $A \in \mathbb{R}^{nXm}$ on top of one another.

$vecs(C) = [c_{11}, 2c_{12}, \ldots, 2c_{1m}, c_{22}, 2c_{23}, \ldots, 2c_{m-1,m}, c_{mm}]^T \in \mathbb{R}^{m(m+1)/2}$.

I_n stands for the n x n identity matrix.

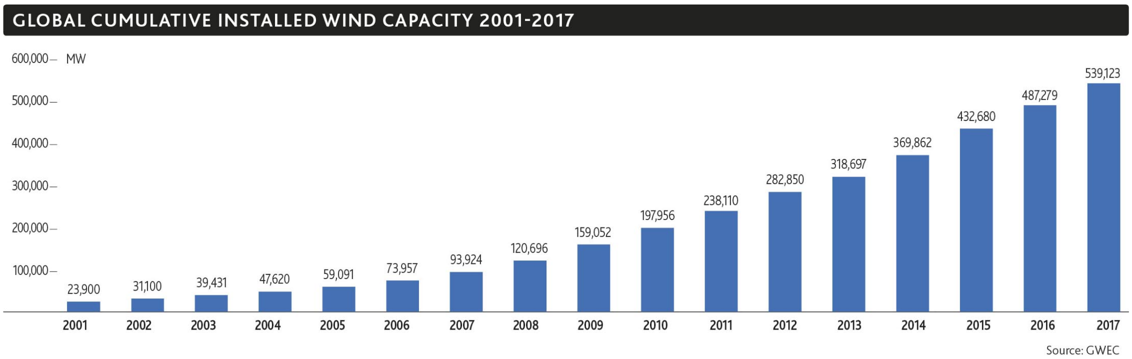
$\nabla f(x)$ represents the gradient of the function $f(x)$, i.e:$\frac{\partial f}{\partial x}$.

A control law is also called a policy, and it is said to be globally asymptotically stabilizing if under the policy, the closed-loop system is Globally Asymptotically Stable (GAS) at an equilibrium.

Chapter 1

Introduction

Recently, there has been an increase in the production and usage of power from the renewable energy resources. This argument is supported by Figure 1-1 which represents the increase in the number of global installed wind capacity. Similarly, there is an increase forecasted in the use of other renewable energy resources such as hydro power, solar power, etc. [7]. It is inevitable that these energy sources have uncertain power production patterns [8]. This uncertainty demands power management using generators. But generators might not be able to operate around the working point which might lead to low efficiency for the system [9].


GLOBAL CUMULATIVE INSTALLED WIND CAPACITY 2001-2017

Source: GWEC

Figure 1-1: Global cumulative installed wind capacity 2001-2017 [1]

In addition to conventional generators, there are other options to attain the same objective of power management. These options include flywheels/governors, distributed energy transitions and Demand Side Management (DSM). The DSM is an interesting topic to study for the following reasons

A large amount of power demand on the grid lines is caused by Thermostatically Controlled Load (TCL). This statement is supported by Figure 1-2 which shows the per-capita increase in the use of air conditioner which is an element of TCL. Also, researchers have identified candidate loads for DSM which are largely TCL.[10, 11, 12, 13, 14, 15, 16]

The TCL has a slack term on their system dynamics which makes it possible to control[17].

The problem statement on DSM of TCL can be divided into two phases.

- Modeling phase where accurate linear/non-linear models are developed using empirical laws
- Control phase where control algorithms are developed/employed to make the TCL track a required a certain power based on the supply.

From a *Systems and Control* perspective, the control phase is interesting to study since it can be posed as a trajectory tracking problem for the system.

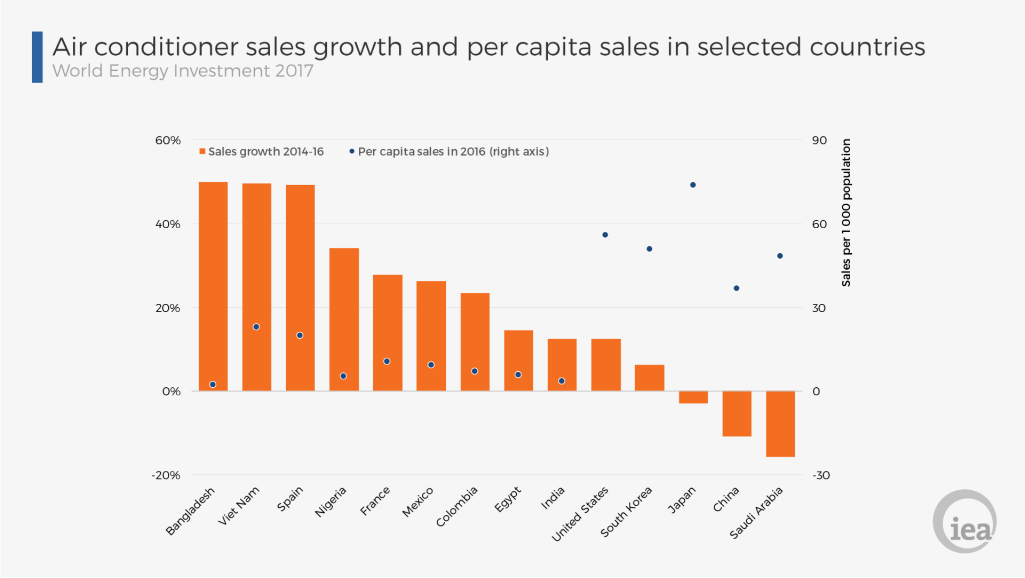


Figure 1-2: Global increase in residential use of TCL - air conditioners[2]

1-1 Related work - Modeling of TCL

This problem of modeling and control has been studied in the literature and some noteworthy articles are mentioned in this section.

[4] develops a Continuous Time (CT) Linear Time Invariant (LTI) state space model. The model relates the offset applied to the temperature set-point of the homogeneous population of TCL (input) to the power consumed by the population (output). It uses the probabilities of a TCL being in an ON/OFF state to calculate the transfer function from which the state space is derived. It also uses an observer based Linear Quadratic Regulator (LQR) controller to achieve the reference tracking objective stated before.

[5] develops a CT bi-linear state space model, relating the offset applied to the temperature set-point of the homogeneous population of TCL (input), to the power consumed by the population (output). The modeling approach used here is based on state bins. Due to its

bi-linear nature, a non-linear controller is developed and uses Lyapunov method for stability analysis.

[18] develops a Discrete Time (DT) LTI state space representation for a heterogeneous population of TCL. It controls the aggregate power (output) by switching the TCLs ON/OFF prematurely but staying within the temperature slack. The modeling approach used here is similar to [5] except for the fact that the bi linearity is removed and is included as a separate block as a part of manual control. It uses an Model Predictive Control (MPC) as the control algorithm.

[19] proposes a 2-dimensional state bin model instead of 1 dimensional model as used in the previous works. It also considers a DT linear time varying state space model which accounts for more uncertainties (like indoor air temperature, second-order dynamics for the TCL) in the model in comparison to the previous works. It uses an MPC as the control algorithm.

From a purely modeling perspective, [20] derives a very detailed model for heterogeneous population of TCL. The modeling is completely based on statistical physics. The importance of this article is that it considers a very huge class of perturbations in its model. This article also suggests how important the role of TCL is, in the energy management of power grids.

A heterogeneous group of TCL consisting of smaller groups of homogeneous TCL is considered in [21]. It proposes a hybrid partial differential equation-based model with numerical stability analysis based on state and output feedback algorithms. It is to be noted that the model is still non-linear.

[22] proposes a model-free control of TCL connected to a district heating network. It uses the following approach. *(i)* Collects the states of the TCL over a period of time *(ii)* determines an offline control action based on the collected states *(iii)* updates the control. It is to be noted that the learning of the control action is based on DT Q-learning approach. The disadvantage of using such a learning algorithm is that the range space must be discretized and can lead to high computational effort or low accuracy. A summary of the literature survey of TCL models are described in Table 1-1.

Literature	Description	TCL population	Control
[4]	CT, Linear	Homogeneous	LQR
[5]	CT, bi-linear	Homogeneous	non-linear
[18]	DT, Linear	Heterogeneous	MPC
[19]	DT, 2D,Linear	Homogeneous with uncertainties	MPC
[20]	CT, probabilistic model	Heterogeneous	-
[22]	DT, non-linear	Heterogeneous	Q-learning

Table 1-1: TCL models - survey

As it can be seen from this survey, the controllers used in the past literature for energy management of TCL are MPC or LQR for a linear model and state/output feedback controllers for the non-linear model. Figure 1-3 and Figure 1-4 represents the control achieved by using an MPC for temperature dependent loads. These controllers depend on the state space model to calculate the control input. Such control algorithms can also be referred to as model-based control which requires an exact system model. A system model can be hard to obtain [20] or produces a low efficient control input when approximate models are used. This leads for to an opportunity to study how model-free control algorithms applies to the DSM problem. The recent advancements in literature in the field of model-free adaptive optimal algorithm is explored in the upcoming section.

1-2 Related work - Adaptive optimal control

Adaptive optimal control originates from reinforcement learning. This origin is explained in [23]. It explains the solution of DT systems and CT systems. The solution for DT systems is direct but the solution of CT is based on certain approximations.

[24] develops a Policy Iteration (PI) algorithm for a CT LTI systems. It solves the regulation problem but only using partial knowledge of the system dynamics. It requires only the knowledge of B (input) matrix. It also provides a lower limit for the sampling time of the CT system to collect the states such that the numerical problem is well-posed. [25] uses the same algorithm to solve a tracking problem instead of a regulation problem. [26] develops a PI algorithm but a CT bi-linear system.

[27] develops Value Iteration (VI) algorithm for CT LTI systems. The difference in this article is that the algorithm gets rid of the assumption on the partial knowledge of the system. It is an iterative algorithm based on the same set of data collected between certain time intervals. It also provides a rigorous stability analysis.

[28] works on the same regulation problem as of [27], but is based on stochastic approximation to develops a VI algorithm. [29] develops a VI algorithm based on stochastic approximation but for uncertain interconnected systems.

The previously mentioned reviewed algorithms are based on state measurements. But [30] and [31] concentrates on output feedback rather than state feedback. [30] focuses on regulation whereas [31] focuses on tracking problem. Both these algorithm are VI algorithms.

The algorithms stated previously are developed for LTI systems assuming that there are no disturbances present. [6]-Chapter 5 develops adaptive algorithm for LTI CT systems with matched and unmatched disturbances. The importance of such algorithms has been studied by applying these algorithms to real-time large-scale system such as power systems in [32] and [33].

VI and PI algorithms for nonlinear affine and non-affine systems with and without disturbances are developed in [6]. It also applies the same to practical system such as inverted pendulum, car suspension systems, etc. A summary of the adaptive optimal control algorithms are described in Table 1-2.

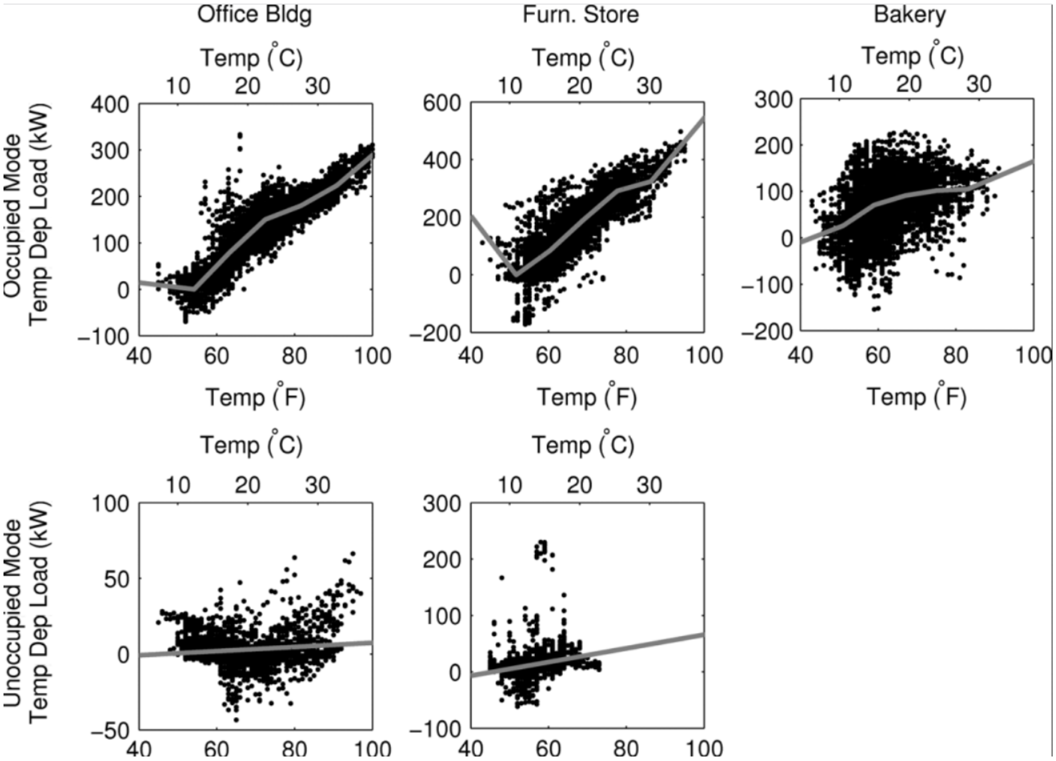


Figure 1-3: Actual (dots) and predicted (line) temperature-dependent load. May - September 2008 [3]

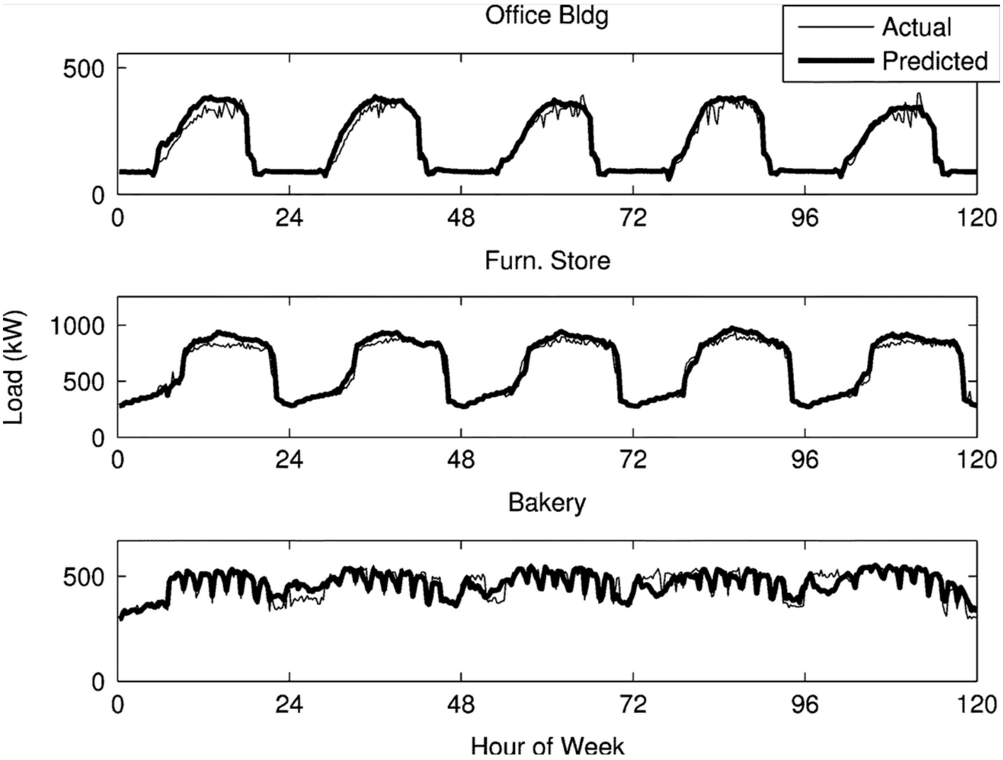


Figure 1-4: Actual versus predicted load time series for Monday-Friday, June 2 to 6, 2008 [3]

Literature	System description	Algorithm description
[25]	Input matrix known	Policy Iteration
[27]	Completely unknown	Value Iteration
[24]	Completely unknown	Policy Iteration
[28]	Completely unknown	Stochastic approximation
[31]	Completely unknown	Output feedback
[34]	With ISS disturbances	Policy Iteration
[6]	Partially linear system	Value Iteration

Table 1-2: Adaptive optimal algorithms - survey

1-3 Research Question (RQ)

The RQs that have been left open in the literature and will be answered in this research thesis are stated below.

RQ 1: Can adaptive optimal control be applied to TCL? Which TCL models should be used?

RQ 2: What are its limitations? How can these limitations be overcome?

RQ 3: How does the parameter heterogeneity affect the performance of the control?

1-4 Report structure

This thesis report is organized as follows.

Chapter 2 works on applying adaptive algorithms based on state feedback, to the homogeneous model developed in [4]. Power tracking using completely unknown and partially known systems dynamics are studied.

Chapter 3 concentrates on applying the output feedback algorithm for the homogeneous model [4]. An indirect adaptive control based PI algorithm is proposed with stability analysis.

Chapter 4 aims on applying non-linear control algorithms for a homogeneous and a heterogeneous population of TCL models developed in [5] and [18] respectively.

Chapter 5 provides detailed discussion, results and comparison. Finally, in Chapter 6, a conclusion is provided also discussing the future works.

Chapter 2

Homogeneous Model - State Feedback (SFB)

This chapter concentrates on discussing the state space model developed in [4], and the algorithms stated in [25] and [27] is applied to the same. It aims in answering the Research Question (RQ) 1 and RQ 2. It states the limitation at the end of the chapter and the upcoming chapter aims at solving these limitations.

2-1 Primary Thermostatically Controlled Load (TCL) model

The behaviour of temperature $\theta(t)$ in a Thermostatically Controlled Load (TCL) is given by

$$\dot{\theta} = \begin{cases} -\frac{1}{CR}(\theta - \theta_{amb} + PR), & \text{ON State.} \\ -\frac{1}{CR}(\theta - \theta_{amb}), & \text{OFF State.} \end{cases} \tag{2-1}$$

where

$$\text{TCL switches from OFF to ON State if } \theta > \theta_s + \frac{\Delta}{2}$$
$$\text{TCL switches from ON to OFF State if } \theta < \theta_s - \frac{\Delta}{2}$$

θ is the TCL temperature - °C,

C is the thermal capacitance - kWh/°C,

R is the thermal resistance - °C/kW,

θ_{amb} is the ambient temperature - °C,

θ_s is the temperature set-point - °C,

Δ is the temperature dead-band - °C,

P is the power drawn - kW,

δ is the step change applied to the input - °C.

For a homogeneous group of TCL, let the steady state distribution of the loads in the ON and OFF states be represented by N_c and N_h. This distribution is proportional to the cooling and the heating time periods T_c and T_h respectively and this relation is given by [4] as follows.

$$N_c = \frac{T_c}{T_c + T_h} N, \quad N_h = \frac{T_h}{T_c + T_h} N$$

Similarly, the number of ON/OFF loads in a given temperature band $[\theta, \theta_+]$ and $[\theta_-, \theta]$ is proportional to the cooling/heating times and given by

$$n_c(\theta) = \frac{N}{T_c + T_h} t_c(\theta), \quad n_h(\theta) = \frac{N}{T_c + T_h} t_h(\theta)$$

respectively. Let the ON/OFF probability density function denoted by $f_1(\theta)$ and $f_0(\theta)$ respectively and the corresponding probability distribution function by $F_1(\theta)$ and $F_0(\theta)$ respectively. These relations are represented in (2-2) and (2-3).

$$f_0(\theta) = \frac{CR}{(T_c + T_h)(\theta_{amb} - \theta)} \tag{2-2}$$

$$f_1(\theta) = \frac{CR}{(T_c + T_h)(PR + \theta_{amb} - \theta)} \tag{2-3}$$

Where

N is the total number of TCLs present and $N = N_c + N_h$,

$t_c(\theta)$ is the time taken to cool down from from θ_+ to $\theta \geq \theta_-$,

$t_h(\theta)$ is the time taken to rise the loads temperature from θ_- to $\theta \leq \theta_+$.

Now, assuming that a step change is made in the set-point (δ) of the TCL, the dead-band (Δ) changes as shown in Figure 2-1. The original dead-band was from θ_-^0 to θ_+^0. After the step change, the dead-band becomes θ_- to θ_+. We now consider four different TCL initial conditions (a, b, c and d as in Figure 2-1) before a step change is applied to the input. The change in average power consumption in regards to this step change in set-point is calculated by integrating the product of the probability density functions (2-2), (2-3) and the Laplace transformed power waveform corresponding to the points a, b, c and d as in Figure 2-1. That is, let $G_a(s), G_b(s), G_c(s)$ and $G_d(s)$ be the Laplace transforms of power waveform corresponding to points a, b, c and d as in Figure 2-1. Then, the average power consumption is given by

$$P_{avg}(s) = P_a(s) + P_b(s) + P_c(s) + P_d(s).$$
$$P_a(s) = \int_{\theta_-}^{\theta_{+0}} f_0(\theta_a) G_a(s) d\theta_a$$
$$P_b(s) = \int_{\theta_-}^{\theta_{+0}} f_1(\theta_b) G_b(s) d\theta_b$$
$$P_c(s) = \int_{\theta_{-0}}^{\theta_-} f_0(\theta_c) G_c(s) d\theta_c$$
$$P_d(s) = \int_{\theta_{-0}}^{\theta_-} f_1(\theta_d) G_d(s) d\theta_d$$

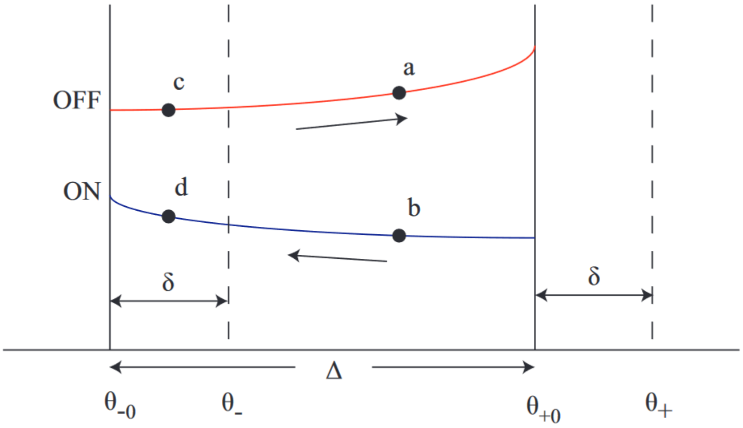


Figure 2-1: Dead-band shift after set-point change [4]

$$\Delta << (\theta_s - \theta_{amb} + PR) \tag{2-4}$$
$$\Delta << (\theta_{amb} - \theta_s) \tag{2-5}$$
$$\delta << \Delta \tag{2-6}$$

Under the assumptions stated in (2-4)-(2-6), the linear transfer function relating the step input change and total power output (P_{tot}) is

$$T(s) = \frac{P_{tot}(s)}{\delta/s} = -d + \frac{A_\Delta \omega s}{s^2 + \omega^2}. \tag{2-7}$$

Where

$$A_\Delta = \frac{5\sqrt{15}C(\theta_{amb} - \theta_+)(PR - \theta_{amb} + \theta_+)}{\eta(P^2R^2 + 3PR(\theta_{amb} - \theta_+) - 3(\theta_{amb} - \theta_+)^2)^{3/2}} \frac{(3PR - \theta_{amb} + \theta_+)N}{T_{c0} + T_{h0}},$$
$$\omega = \frac{2\sqrt{15}C(\theta_{amb} - \theta_+)(PR - \theta_{amb} + \theta_+)}{CR\Delta\sqrt{(P^2R^2 + 3PR(\theta_{amb} - \theta_+) - 3(\theta_{amb} - \theta_+)^2)}},$$
$$d = \frac{N}{\eta R}.$$

Here

θ_+ and θ_- can be inferred from Figure 2-1,

σ is the damping factor,

η is the thermal efficiency of the TCL,

T_{c0} is the steady state cooling time before a step change is applied,

T_{h0} is the steady state heating time before a step change is applied.

The corresponding state space representation of the transfer function (2-7) is

$$\dot{x} = \underbrace{\begin{bmatrix} -2\sigma & -\omega \\ \frac{\sigma^2+\omega^2}{\omega} & 0 \end{bmatrix}}_{\mathbf{A}} x + \underbrace{\begin{bmatrix} \omega A_\Delta \\ 0 \end{bmatrix}}_{\mathbf{B}} u \tag{2-8}$$

$$y = \underbrace{\begin{bmatrix} -1 & 0 \end{bmatrix}}_{\mathbf{C}} x + \underbrace{-d}_{\mathbf{D}}\, u \tag{2-9}$$

2-2 Problem formulation

The goal of the optimal trajectory tracking problem is to find the optimal control policy u^* so as to make the system (2-8) - (2-9) track a desired reference trajectory (y_d) by minimizing the predefined cost function

$$J = \frac{1}{2}\int_0^\infty ((y - y_d)^T Q(y - y_d) + u^T Ru)dt$$

($Q \geq 0, R > 0$) and stabilizes the system.

2-3 Conventional Algebraic Riccati Equation (ARE) solution

Let us define an augmented state space matrix as stated below.

$$\begin{bmatrix} \dot{x} \\ \dot{y}_d \end{bmatrix} = \begin{bmatrix} A_2 & 0 \\ 0 & F \end{bmatrix} \begin{bmatrix} x \\ y_d \end{bmatrix} + \begin{bmatrix} B_2 \\ 0 \end{bmatrix} u \equiv TX + B_1 u. \tag{2-10}$$

$$y = \begin{bmatrix} C_2 & -1 \end{bmatrix} \begin{bmatrix} x \\ y_d \end{bmatrix} \equiv C_1 X$$

Where F represents the command generator dynamics ($\dot{y}_d = Fy_d$) which generates the trajectory to be followed by the TCL output. In the augmented system representation, the output y represents the error term $Cx(t) - y_d$ and therefore the objective of the problem becomes to find a control input $u_1 = K_1 X$ such that

$$\lim_{t\to\infty} y(t) \to 0 \quad \text{and} \quad J = \frac{1}{2}\int_0^\infty (X^T C_1^T Q C_1 X + u^T Ru)dt$$

is minimized. Here, D is assumed to be zero but this assumption is validated in section 2.5.

Assumption 1-$\mathcal{A}$. *System is controllable and observable*

To find the solution of the above minimization problem, using ARE, the system should be both controllable and observable. The system (2-8)-(2-9) satisfies the assumption 1-$\mathcal{A}$ but the system (2-10) does not. Hence a discounted cost is considered as represented in (2-11). The solution of the Lyapunov value function minimizing this cost function can be found by solving the conventional ARE as stated in (2-12). The solution of the Lyapunov value function can be related to the state feedback gain by (2-13). Although this method solves the problem, this method uses the system matrices (A, B and C). Hence an algorithm to find the solution of the minimization problem (2-11) without using the system dynamic matrix A is described in the next section[1].

$$J(X(t)) = V(X(t)) = \frac{1}{2}\int_t^\infty e^{-\gamma(\tau-t)}(X(t)^T C_1^T Q C_1 X(t) + u^T R u)d\tau \tag{2-11}$$

$$(T - 0.5\gamma I)^T P + P(T - 0.5\gamma I) - PB_1R^{-1}B_1^T P + C_1^T Q C_1 = 0 \tag{2-12}$$

$$K_1 = -R^{-1}B_1^T P \tag{2-13}$$

2-4 Integral Reinforcement Learning (IRL) for partially known system

To remove the need for system dynamic matrix A, a learning algorithm is proposed [25]. In a time interval Δt, (2-11) can be approximated as (2-14). If a quadratic positive definite symmetric solution P exist for the value function, (2-14) can be rewritten as (2-15). This leads to the online Algorithm-1.

$$V(X(t)) = \frac{1}{2}\int_t^{t+\Delta t} e^{-\gamma(\tau-t)}(X(t)^T C_1^T Q C_1 X(t) + u^T R u)d\tau + e^{-\gamma t}V(X(t+\Delta t)) \tag{2-14}$$

$$X(t)^T P X(t) = \frac{1}{2}\int_t^{t+\Delta t} e^{-\gamma(\tau-t)}(X(t)^T C_1^T Q C_1 X(t) + u^T R u)d\tau + e^{-\gamma t}X(t+\Delta t)^T P X(t+\Delta t) \tag{2-15}$$

2-5 Results

The system represented in (2-8) - (2-9) (also represented briefly in (2-16) -(2-17)) contains the term which describes the system dynamics of the TCL in the matrices B and D. This is not favored because the matrix B is supposed to be known in the Algorithm-1 and the objective of the algorithm is to find the optimal state feedback gain without knowing this system dynamics. Hence, a stable input filter as described in (2-18) is applied to the input and an indirect control input v is created.

$$\dot{x} = Ax + Bu \tag{2-16}$$

$$y = Cx + Du \tag{2-17}$$

$$\text{Let } \dot{v} = Eu + Gv \tag{2-18}$$

[1] Here P represents the solution of the ARE.

Algorithm 1: Policy Iteration (PI) algorithm for Linear Quadratic Tracking (LQT)

```
**Result:** Riccati solution $P$
1 **Input**: A initial stabilizing control policy

2 **Initialization**: Start with an admissible control input $u^0 = -K^0 x$, $i \leftarrow 0$
3 **Policy evaluation**: Solve for $P^i$ from (2-15)
4 **Policy evaluation**: Update the control policy using $u^{i+1} = -R^{-1}B_1^T P^i X$
5 **Stopping criterion**: Let $i \leftarrow i+1$ and go to **Step 3**, until
      $||P_i - P_{i-1}|| \leq \epsilon$
  where $\epsilon > 0$ is sufficiently small predefined threshold.
```

Applying (2-18) in (2-16)-(2-17) yields the system (2-19)-(2-20) where the filter coefficients E and G are design parameters.

$$\begin{bmatrix} \dot{x} \\ u \end{bmatrix} = \underbrace{\begin{bmatrix} A & B \\ 0 & E \end{bmatrix}}_{A_2} \begin{bmatrix} x \\ u \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ G \end{bmatrix}}_{B_2} v \tag{2-19}$$

$$y = \underbrace{\begin{bmatrix} C & D \end{bmatrix}}_{C_2} \begin{bmatrix} x \\ u \end{bmatrix} \tag{2-20}$$

Now, (2-19)-(2-20) has a structure that can be used to apply Algorithm-1. For the simulation purpose, the design choices made [35] are given in Table 2-1. Algorithm-1 is applied to the system (2-19) - (2-20). The convergence of the matrix norm to the nominal values found by solving (2-12) is shown in Figure 2-2. The solution obtained by solving the LQT ARE (2-12) and the solution obtained from Algorithm-1 are shown below.

Parameter	Value	Parameter	Value
Power (P)	6kW	R	0.12 $°C/kW$
C	3.6 $kWh/°C$	N	100
η	0.5	σ	0.002 $hours^{-1}$
θ_{set}	20 $°C$	θ_{amb}	32 $°C$
Δ	1 $°C$	$[E\ F]$	$[-1\ -1]$
$[R\ Q]$	[1 5]	γ	0.1

Table 2-1: Design parameters

ARE solution

$$P^* = \begin{bmatrix} 0.1396 & -0.0010 & 2.2327 & 0.1396 \\ -0.0010 & 0.0013 & -0.0048 & -0.0019 \\ 2.2327 & -0.0048 & 36.2917 & 2.2329 \\ 0.1396 & -0.0019 & 2.2329 & 0.1411 \end{bmatrix} \quad K^* = \begin{bmatrix} -2.2327 & 0.0048 & -36.2917 & -2.2329 \end{bmatrix}$$

Solution from Algorithm-1

$$P = \begin{bmatrix} 0.1395 & -0.0010 & 2.2331 & 0.1396 \\ -0.0010 & 0.0011 & -0.0043 & -0.0020 \\ 2.2331 & -0.0043 & 36.3031 & 2.2334 \\ 0.1396 & -0.0020 & 4.4669 & 0.0706 \end{bmatrix} \quad K = \begin{bmatrix} -2.2331 & 0.0043 & -36.3031 & -2.2334 \end{bmatrix}$$

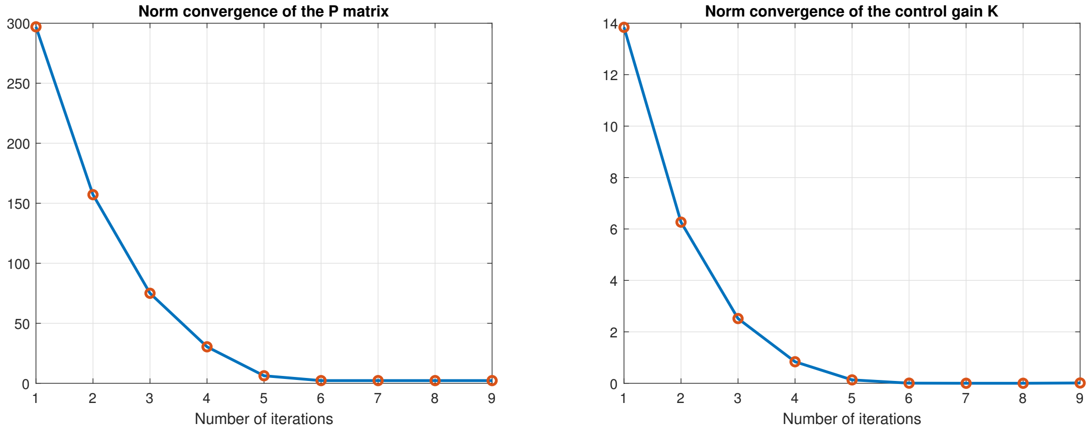


Figure 2-2: Norm convergence of matrices P and K to their optimal values - Algorithm-1

2-6 IRL for completely unknown system

Considering a Continuous Time (CT)Linear Time Invariant (LTI) system as in (2-16) - (2-17), the objective of this section is to find the solution to problem 2-2 without knowing the system dynamics. To begin with, the system notation is rewritten in the form (2-21) where $A_k = A - BK^i$. For a completely known system controlled using the state feedback gain (K_k), the Kleinman algorithm [36] gives (2-22) and (2-23). Applying (2-22) and (2-23) in (2-21) yields (2-24). It can be noted that (2-24) does not involve the system dynamic matrices and can be solved by only using the state measurements and the input over a period of time.

$$\dot{x} = A_k x + B(K^i x + u) \tag{2-21}$$

$$0 = (A - BK^i)^T P^i + P^i (A - BK^i) + Q + K^{iT} R K^i \tag{2-22}$$

$$K^i = R^{-1} B^T P^{i-1} \tag{2-23}$$

$$x(t+\Delta t)^T P x(t+\Delta t) - x(t)^T P x(t) = \\ -\int_t^{t+\Delta t} x(t)(Q + K^{iT} R K^i) x(t) d\tau + 2\int_t^{t+\Delta t} (u + K^i x)^T R K^{i+1} x d\tau \tag{2-24}$$

Now, let us make the following notations

$$\hat{P} = vecs(P), \quad \bar{x} = x \otimes x,$$

$$I_{xx} = \Big[\int_{t_0}^{t_1} x \otimes x d\tau, \int_{t_1}^{t_2} x \otimes x d\tau, \cdots \int_{t_{l-1}}^{t_l} x \otimes x d\tau\Big]^T,$$

$$I_{xu} = \Big[\int_{t_0}^{t_1} x \otimes u d\tau, \int_{t_1}^{t_2} x \otimes u d\tau, \cdots \int_{t_{l-1}}^{t_l} x \otimes u d\tau\Big]^T,$$

$$\delta_{xx} = \Big[\bar{x}(t_1) - \bar{x}(t_0), \bar{x}(t_2) - \bar{x}(t_1), \ldots, \bar{x}(t_{l-1}) - \bar{x}(t_l)\Big]^T,$$

where $0 \leq t_0 < t_1 < \ldots t_l$. Then for any stabilizing gain K^i implies the following linear consistent set of equation (2-25) which can be solved for $\hat{P}, K^{i+1}$. This leads to the adaptive Algorithm-2. It is to note that this algorithm solves regulation problem and NOT a tracking problem.

$$\begin{bmatrix} \hat{P}^i \\ vec(K^{i+1}) \end{bmatrix} = (\Theta_k^T \Theta_k)^{-1} \Theta_k^T E_k \tag{2-25}$$

where

$$\Theta_k = \Big[\delta_{xx}, -2I_{xx}(I_n \otimes K^{iT} R) - 2I_{xu}(I_n \otimes R)\Big]^T \quad E_k = -I_{xx} vec(Q_k)$$

Algorithm 2: Value Iteration (VI) algorithm for Linear Quadratic Regulator (LQR)

```
  Result: Riccati solution $P, K$
1 Input: A initial stabilizing control policy

2 Initialization: Start with an admissible control input $u^0 = -K^0 x$, $k \leftarrow 0$
3 Online data collection: Apply the control policy $u = -K^0 x + e$ and collect the
  system output and input information. construct the matrix $\Theta_k$ and $K_k$
4 Policy evaluation: Solve for $P, K$ from (2-25)
5 Stopping criterion: Let $k \leftarrow k + 1$ and go to Step 3, until

                    $||P_k - P_{k-1}|| \leq \epsilon$

  where $\epsilon > 0$ is sufficiently small predefined threshold.
6 Actual control policy improvement: Terminate the exploration noise $e$ and apply
  the control policy $u = K^k x$.
```

2-7 Results

Since Algorithm-2 aims in regulation rather than tracking, a Linear Quadratic Integral (LQI) problem is posed as in (2-26). Here the matrices A, B and C are in accordance with (2-19)-(2-20) and the augment matrix as in (2-26) is used in Algorithm-2.

$$\begin{bmatrix} \dot{x} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u, \quad y = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix} \tag{2-26}$$

The design parameter used for simulation are similar to Table 2-1. The data matrices (Θ_k, E_k) are collected as mentioned in the algorithm for a time period of 5 seconds. During this period, an exploration a noise of the form (2-27) is applied to the system. The norm convergence of the gain matrices while solving the recursive equation (2-25) is shown in Figure 2-3. The solution obtained by solving the LQT ARE (2-12) and the solution obtained from Algorithm-1 are also shown below.

$$u = \sum_{\omega=1}^{100} sin(\omega t) \tag{2-27}$$

ARE solution

$$P^* = \begin{bmatrix} 0.1440 & -0.0002 & 2.2960 & 0.1363 \\ -0.0002 & 0.0059 & -0.0047 & -0.0001 \\ 2.2960 & -0.0047 & 37.3273 & 2.2361 \\ 0.1363 & -0.0001 & 2.2361 & 0.1345 \end{bmatrix} \quad K^* = \begin{bmatrix} -2.2960 & 0.0047 & -37.3273 & -2.2361 \end{bmatrix}$$

Solution from Algorithm-2

$$P = \begin{bmatrix} 0.1440 & -0.0001 & 2.2960 & 0.1363 \\ -0.0001 & 0.0059 & -0.0047 & -0.0001 \\ 2.2960 & -0.0047 & 37.3273 & 2.2361 \\ 0.1363 & -0.0001 & 2.2361 & 0.1345 \end{bmatrix} \quad K = \begin{bmatrix} -2.2960 & 0.0047 & -37.3273 & -2.2361 \end{bmatrix}$$

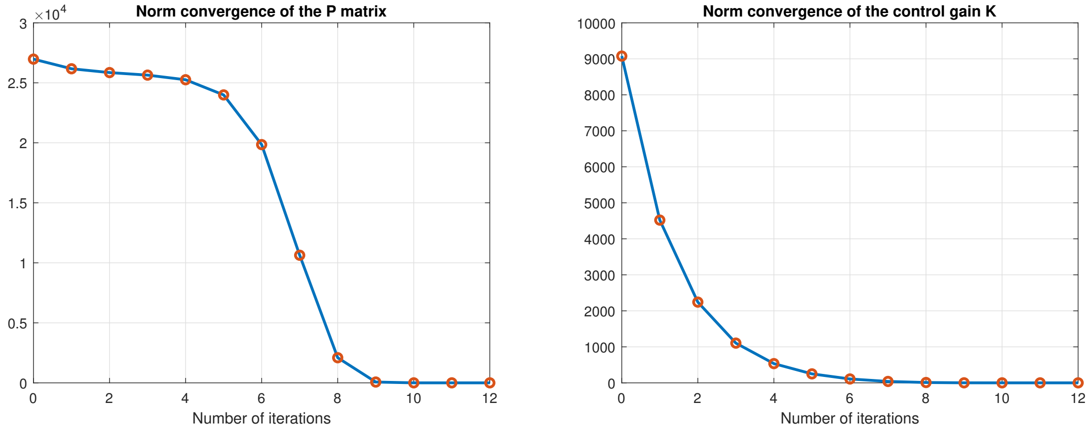


Figure 2-3: Norm convergence of matrices P and K to their optimal values - Algorithm-2

2-8 Discussion

The Algorithm-1 and Algorithm-2 is implemented for the system represented in (2-19)-(2-20). The significant difference between these two algorithms is that

> Algorithm-1 is *PI for partially known system* and Algorithm-2 is *VI for completely unknown system*.
>
> Algorithm-1 solves a tracking problem and Algorithm-2 solves a regulation problem.
>
> A least squares problem is solved in Algorithm-1 whereas the solution is obtained by recursion in Algorithm-2. Due to this key difference, Algorithm-2 is computationally less intensive.
>
> During this implementation, the computationally complexity is calculated in terms of the time consumed for computation. (2-25) attains the solution in $43.6601ms$ whereas (2-15) attains the solution in $96.324ms$ which is in accordance with expectations.

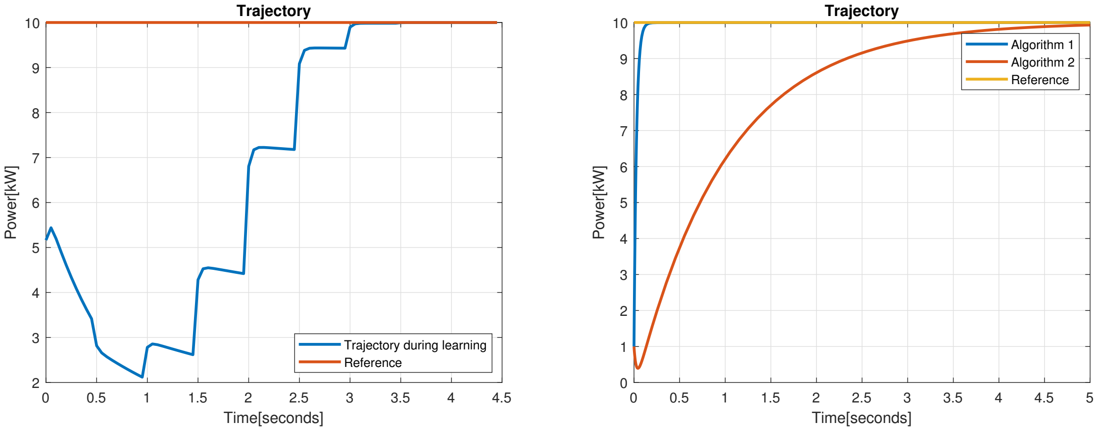


Figure 2-4: Output trajectories

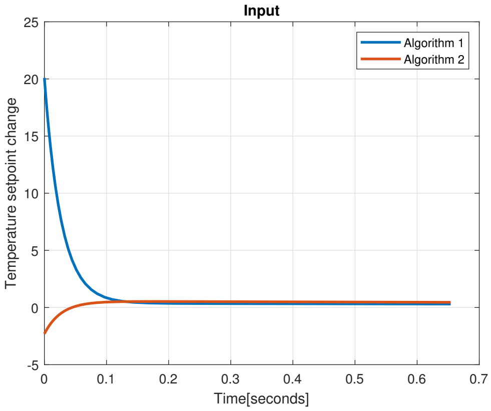


Figure 2-5: Control inputs

The system trajectory during the learning phase for Algorithm-1 and the system trajectories after the learning process with an initial state state of $x_0 = \begin{bmatrix} -1 & 1 & 0 \end{bmatrix}^T$ is shown in in Figure 2-4. It can be inferred that Algorithm-1 converges much faster than its counterpart. But this is accounted for the difference in the magnitude of input as can be seen from Figure 2-5. The convergence rate is a trade of to be made with input magnitude applied to the system. To apply this control algorithm, in reality, the states have to be measured from the real world. But a physical interpretation of the states for the system (2-8)-(2-9) cannot be found in the literature. Hence, although the algorithm works from a control perspective, the implementation aspect of it falls short. An alternative solution to this problem can be proposed in the following ways

- Adopt a Output Feedback (OPFB) algorithm since the output $y(t)$ for the system is measurable.
- Adopt a different system representation where the states are completely measurable.

These solution aspects are investigated in the upcoming chapters.

Chapter 3

Homogeneous Model - Output Feedback (OPFB)

In line with the arguments stated in Chapter 2, an algorithm where the output measurements are used instead of state measurements is necessary. The Research Question (RQ) 2 is partly answered in this chapter where the OPFB algorithm is applied to the system and its performance is studied.

3-1 Model-free OPFB algorithm

Similar to the Chapter 2, let us assume that a state feedback control of the form $u = Kx(t)$ is used. Applying this to the system (2-8)-(2-9), the solution $x(t)$ of the system becomes (3-1). A generalised form of this equation can be written as (3-2). The solution $y(t)$ in terms of $x(t)$ can be written as (3-3). Using this representation, suppose that there are N output measurements available, (3-4) can be constructed.

$$x(t) = e^{(t-t_0)(A+BK)}x(t_0) \tag{3-1}$$

$$x(t-i\Delta t) = e^{-i\Delta t(A+BK)}x(t) \tag{3-2}$$

$$y(t-i\Delta t) = Ce^{-i\Delta t(A+BK)}x(t) \tag{3-3}$$

$$\underbrace{\begin{bmatrix} y(t) \\ y(t-\Delta t) \\ \vdots \\ y(t-(N-1)\Delta t) \end{bmatrix}}_{\bar{y}_t} = \underbrace{\begin{bmatrix} C \\ Ce^{-\Delta t(A+BK)} \\ \vdots \\ Ce^{-(N-1)\Delta t(A+BK)} \end{bmatrix}}_{G} x(t) \tag{3-4}$$

$$\implies \bar{y}_t = Gx(t) \tag{3-5}$$

Here, $x(t) \in \mathbb{R}^n$, $y(t) \in \mathbb{R}^1$, $G \in \mathbb{R}^{N\times n}$. We now have the output measurement in terms of the state measurements. The objective is to find a stabilizing control input $u(t)$ such that

$$\lim_{t\to\infty} y(t) - y_d \to 0$$

and minimizing the predefined cost function (3-6).

$$V(t) = \int_t^\infty e^{-\gamma(\tau-t)}\Big(y(t)^T Qy(t) + u(t)^T Ru(t)\Big)dt \tag{3-6}$$

Next, we try to learn the solution (P) of the value function in terms of the output measurements. The quadratic value function whose solution is to be found is represented in (3-7). Using (3-5) in this equation results in (3-8) where $G_N = (G^T G)^{-1}G^T \in \mathbb{R}^{n\times N}$. This is valid since the assumption 1-$\mathcal{A}$ is satisfied.

$$V(t) = x(t)^T Px(t) \tag{3-7}$$

$$V(t) = (G_N\bar{y}_t)^T P(G_N\bar{y}_t) \tag{3-8}$$

$$\implies V(t) = \bar{y}_t^T G_N^T PG_N\bar{y}_t$$

(3-8) can be rewritten as

$$V(t) = \bar{y}_t^T \bar{P}\bar{y}_t$$

where

$$\bar{P} = G_N^T PG_N \in \mathbb{R}^{N\times N}$$

Using (3-1)-(3-8), the equation (2-24) can be rewritten as (3-9). This equation does not require the system state measurements and results in the Algorithm-3.

$$\begin{aligned} e^{-\gamma\Delta t}\bar{y}_{t+\Delta t}^T\bar{P}\bar{y}_{t+\Delta t} - \bar{y}_t^T\bar{P}\bar{y}_t = -\int_t^{t+\Delta t} e^{-\gamma(\tau-t)}\bar{y}_t^T\bar{Q}\bar{y}_t d\tau \\ -2\int_t^{t+\Delta t} e^{-\gamma(\tau-t)}w^T R\bar{K}^{i+1}\bar{y}_t d\tau \end{aligned} \tag{3-9}$$

where $\bar{Q} = \begin{bmatrix}\underline{1} & 0 & 0 & 0\end{bmatrix}^T Q \begin{bmatrix}\underline{1} & 0 & 0 & 0\end{bmatrix}$. The state feedback gain and the output feedback gain can be related by

$$u^i = K^i x = K^i G_N^i \bar{y}_t = \bar{K}^i\bar{y}_t \tag{3-10}$$

3-2 Results and discussion

Consider the system (2-8)-(2-9). The system satisfies assumption 1-$\mathcal{A}$. We seek a controller of the form $u = K\bar{y}_t$. When the Algorithm-3 is applied to this system with $\gamma = 0.1, R = 1, \Delta t = 0.1$, number of stored data in the history as 3, probing noise of the form (2-27), it would result in tracking performance (for different input weights (R)) as shown in the Figure 3-1. It can be inferred that convergence and the input magnitude are inversely proportional in terms of trade-off that must be made to satisfy the performance measure that most suits the application. As Q increases, the input magnitude increases but the convergence time decreases. This can also be inferred from Table 3-1 where the norm of the feedback matrices increases with an increasing weight of Q. A study was also made to infer how the performance changes with an increasing number of data stored. As can be seen from the Table-3-2, the cost decreases with an increasing number of data variables but increases the computational complexity.

```
Algorithm 3: VI algorithm for OPFB
  Result: Riccati solution $\bar{P}, \bar{K}$
1 Input: A initial stabilizing control policy
2 Initialization: Find an initial control policy $u^0$, $i \leftarrow 0$ and $t \leftarrow 0$
3 Online data collection: Apply the control policy $u = u^0 + e$ and collect the system
  output and input information.
4 Policy evaluation: Solve for $\bar{P}_i$ and $\bar{K}_i$ from (3-9)
5 Stopping criterion: Let $i \leftarrow i + 1$ and $t \leftarrow t + \Delta t$, and go to Step 3, until
      $||\bar{P}_i - \bar{P}_{i-1}|| \leq \epsilon$
  where $\epsilon > 0$ is sufficiently small predefined threshold.
6 Actual control policy improvement: Terminate the exploration noise $e$ and
  $u = u_0$ as the control input. Apply the control policy $u = \bar{K}_i \bar{y}_t$.
```

Q	K	$\|\|K\|\|$
2	$\begin{bmatrix} -0.1055 & 0.1737 & -0.0000 & -0.1283 \end{bmatrix}$	0.2404
2.5	$\begin{bmatrix} -0.1251 & 0.2179 & -0.0000 & -0.1530 \end{bmatrix}$	0.2942
3	$\begin{bmatrix} -0.1450 & 0.2619 & -0.0000 & -0.1771 \end{bmatrix}$	0.3478
3.5	$\begin{bmatrix} -0.1626 & 0.3075 & -0.0000 & -0.2054 \end{bmatrix}$	0.4039

Table 3-1: Performance comparison for varying weights Q

N	5	6	7
K^T	$\begin{bmatrix} 0.6902 \\ 0.1111 \\ -0.4557 \\ 0.0644 \\ -0.4114 \end{bmatrix}$	$\begin{bmatrix} 0.4406 \\ 0.2315 \\ -0.0301 \\ -0.2762 \\ 0.0416 \\ -0.4091 \end{bmatrix}$	$\begin{bmatrix} 0.3066 \\ 0.2163 \\ 0.0963 \\ -0.0538 \\ -0.1907 \\ 0.0297 \\ -0.4066 \end{bmatrix}$
Cost	1.1270e+05	1.1134e+05	1.1052e+05

Table 3-2: Performance comparison for varying history length

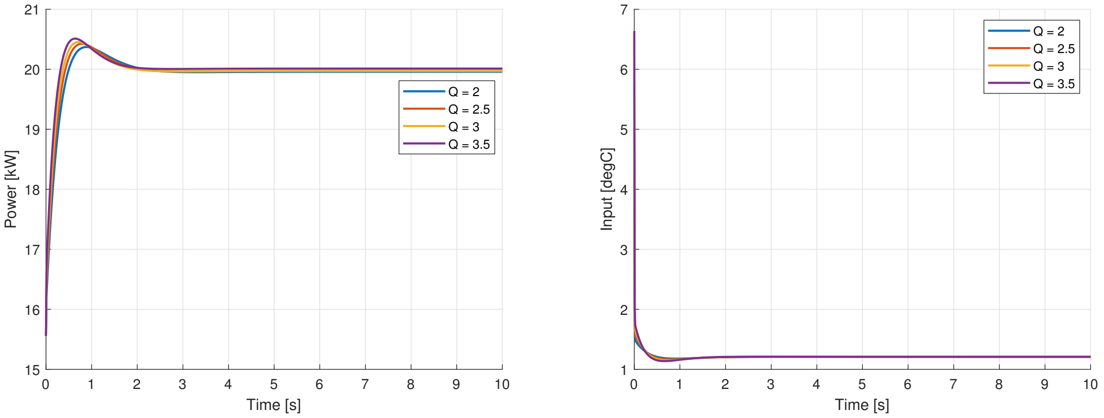


Figure 3-1: Output trajectories and control inputs for varying weights Q (20kW)

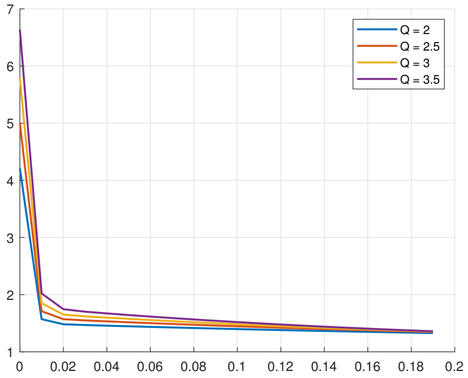


Figure 3-2: Control inputs for varying weights Q

3-3 Proposed algorithm

The Algorithm-3 aims at finding the Riccati solution P for a completely unknown system using VI. But in the case of TCL systems, the input matrix B is easy to be known or an input filter of the form (2-18) can be applied to the system such that B can be made to be known. Also, Policy Iteration (PI) is faster than Value Iteration (VI). Hence a PI algorithm for a partially unknown system is sought for an output feedback control. Let us consider the system with the state space representation (2-8)-(2-9). The conventional solution with the cost (3-6) can be found by solving the Riccati equation offline by knowing the system dynamics as follows

$$(A - 0.5\gamma I)^T P + P(A - 0.5\gamma I) - PBR^{-1}B^T P + C^T QC = 0 \tag{3-11}$$

The Riccati solution can be found without knowing the system dynamics online using the state measurements for a partially unknown system (knowledge of B is required) by the algorithm stated in [25]. The solution is found by solving the equation (3-12) recursively.

$$x(t)^T P^i x(t) - e^{-\gamma\Delta t} x(t+\Delta t)^T P^i x(t+\Delta t) = \frac{1}{2}\int_t^{t+\Delta t} e^{-\gamma(\tau - t)} \Big[x(t)^T C^T Q C x(t) + u_i^T R u_i\Big] d\tau \tag{3-12}$$

Using (3-1)-(3-7) in (3-12) results in (3-13)

$$\bar{y}_t^T \bar{P}_i \bar{y}_t - e^{-\gamma\Delta t} \bar{y}_{t+\Delta t}^T \bar{P}_i \bar{y}_{t+\Delta t} = \frac{1}{2}\int_t^{t+\Delta t} e^{-\gamma(\tau - t)} \Big[\bar{y}_t^T Q \bar{y}_t + u_i^T R u_i\Big] d\tau \tag{3-13}$$

Equation (3-13) is independent of the states. The policy update step is given by $u^i = -R^{-1}B^T G \bar{P}_i \bar{y}_t$. But the equation (3-13) cannot be used to make the algorithm online as the policy update step still is a function of G. Hence an indirect adaptive control algorithm can be adapted where the system is estimated and controlled simultaneously.

Now, a detailed description of the parameter estimation algorithm in Continuous Time (CT) is given here to keep the presentation self-contained as discussed in [37].

Consider a Single Input Single Output (SISO) system with the relation

$$y = G(s)u$$

where

$$G(s) = \frac{Z(s)}{R(s)}$$

where u is the plant input and y is the plant output. Let

$$\begin{aligned} R(s) &= s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 \\ Z(s) &= b_m s^m + b_{m-1}s^{m-1} + \cdots + b_1 s + a_0 \end{aligned}$$

Combining the above two equation, a n^{th} order differential equation can be represented as follows

$$y^{(n)} + a_{n-1}y^{(n-1)} + \cdots + a_1\dot{y} + a_0 y = b_m u^{(m)} + b_{m-1}u^{(m-1)} + \cdots + b_1\dot{u} + b_0 u \tag{3-14}$$

Now, collecting all the parameters to be estimated together, we have

$$\theta^* = \begin{bmatrix} b_m & \dots & b_0 & a_{n-1} & \dots & a_0 \end{bmatrix}^T$$

With the above definition, (3-14) can be re-written as

$$y^{(n)} = \theta^* \begin{bmatrix} u^{(m)} \\ \dots \\ u \\ -y^{(n-1)} \\ \dots \\ -y \end{bmatrix} \tag{3-15}$$

Filtering each side of (3-15) with $\frac{1}{\Delta(s)}$ where $\Delta(s)$ is a monic Hurwitz polynomial of degree n, we obtain the parametric model

$$z = \theta^* \phi$$

where

$$\begin{aligned} z &= \frac{s^n}{\Delta(s)} y \\ \theta^* &= \begin{bmatrix} b_m & \dots & b_0 & a_{n-1} & \dots & a_0 \end{bmatrix}^T \in \mathbb{R}^{n+m+1} \\ \phi &= \begin{bmatrix} \frac{s^m}{\Delta(s)} u & \dots & \frac{1}{\Delta(s)} u & -\frac{s^{n-1}}{\Delta(s)} y & \dots & \frac{1}{\Delta(s)} y \end{bmatrix}^T \end{aligned}$$

Let us define a parametric estimation model given by

$$\hat{z} = \theta(t)^T \phi \tag{3-16}$$

where $\theta(t)$ is the estimate of θ at time t. The estimation error is constructed as

$$\epsilon = \frac{z - \hat{z}}{m_s^2} = \frac{z - \theta^T \phi}{m_s^2} \tag{3-17}$$

Here $m_s^2 = 1 + n_s^2$, where $n_s \geq 0$. An appropriate choice for n_s includes the ones mentioned below. α and P are design parameters chosen to tune the algorithm.

$$n_s^2 = \alpha \phi^T \phi, \quad \alpha > 0 \quad \text{or} \quad n_s^2 = \phi^T P \phi, \quad P = P^T > 0$$

Let us define a cost function as follows

$$J(\theta) = \frac{\epsilon^2 m_s^2}{2} = \frac{z - \theta^T \phi}{2 m_s^2}$$

This cost function is convex in nature and has a global minimum. The parameters can be estimated by a gradient descent method. The gradient of the cost function is given by

$$\nabla J = \frac{\partial J}{\partial \theta} = \frac{(z - \theta^T \phi)}{m_s^2}(-\phi) = -\epsilon \phi$$

The adaptive algorithm to estimate the cost based on the above-mentioned gradient descent method is given by

$$\dot{\theta} = -\Gamma \epsilon \phi \tag{3-18}$$

Here Γ represents a gain which determines the step size which is a tuning variable. The adaptive law (3-18) together with the estimation model (3-16), constitute the gradient parameter identification algorithm. The gradient algorithm has the following properties

- $\epsilon, \epsilon m_s, \dot{\theta} \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\theta \in \mathcal{L}_\infty$
- If $\frac{\phi}{m_s^2}$ is persistently exciting, then $\theta(t) \to \theta^*$
- If the plant has stable poles and no zero-pole cancellations and the input u is sufficiently rich of order $n+m+1$, then ϕ, $\frac{\phi}{m_s^2}$ is persistently exciting and $|\theta(t) - \theta^*|$, ϵ, ϵm_s converges to 0 exponentially fast.

Algorithm 4: PI algorithm for OPFB

```
Result: Riccati solution $\bar{P}$
1 Input: A initial stabilizing control policy

2 Initialization: Apply an initial control policy $u^0$, $i \leftarrow 0$ and $t \leftarrow 0$, $K \leftarrow 0$
3 Online data collection: Apply the control policy $u = u^i + e$ and collect the system
  output and input information.
4 Data estimation: From the data collected, estimate the parameters $\theta(t)$ from (3-18)
  and construct the matrix G as in (3-4).
5 Policy evaluation: Solve for $\bar{P}_i$ from (3-13)
6 Policy improvement: Apply the control policy $u^i = -R^{-1}B^TG\bar{P}_i\bar{y}_t \ = \ \bar{K}\bar{y}_t$
7 Gain estimation: Transform $\bar{K}_i$ to $K_i$ using the relation (3-10)
8 Stopping criterion: Let $i \leftarrow i + 1$ and $t \leftarrow t + \Delta t$, and go to Step 3, until

      $||\bar{P}_i - \bar{P}_{i-1}|| \leq \epsilon$

  where $\epsilon > 0$ is sufficiently small predefined threshold.
```

The proof of these properties is given in section 3.6.1 of [37]. This gradient descent algorithm (3-18) along with the policy evaluation step (3-13) can be together used to propose the output feedback based indirect policy iteration algorithm, Algorithm-4. A flowchart of Algorithm-4 is represented in Figure 3-3.

Lemma 1. *The equation (3-11) and (3-13) converge to the same positive definite solution.*

Proof. Dividing (3-13) by Δt and taking a limit result in

$$\lim_{\Delta t \to 0} \frac{\bar{y}_t^T \bar{P} \bar{y}_t - e^{-\gamma \Delta t} \bar{y}_{t+\Delta t}^T \bar{P} \bar{y}_{t+\Delta t}}{\Delta t} = \lim_{\Delta t \to 0} \frac{\int_t^{t+\Delta t} e^{-\gamma(\tau - t)} \Big[\bar{y}_t^T Q \bar{y}_t + u^T R u\Big] d\tau}{\Delta t}$$

$$\lim_{\Delta t \to 0} \frac{\int_t^{t+\Delta t} e^{-\gamma(\tau - t)} \Big[\bar{y}_t^T Q \bar{y}_t + u^T R u\Big] d\tau}{\Delta t} = \bar{y}_t^T Q \bar{y}_t + u^T R u = x(t)^T C^T Q C x(t) + u^T R u$$

$$\begin{aligned} \lim_{\Delta t \to 0} & \frac{\bar{y}_t^T \bar{P} \bar{y}_t - e^{-\gamma \Delta t} \bar{y}_{t+\Delta t}^T \bar{P} \bar{y}_{t+\Delta t}}{\Delta t} \\ &= \lim_{\Delta t \to 0} \Big(-\gamma e^{-\gamma \Delta t} \bar{y}_{t+\Delta t}^T \bar{P} \bar{y}_{t+\Delta t} + e^{-\gamma \Delta t} \dot{\bar{y}}_{t+\Delta t}^T \bar{P} \bar{y}_{t+\Delta t} + e^{-\gamma \Delta t} \bar{y}_{t+\Delta t}^T \bar{P} \dot{\bar{y}}_{t+\Delta t} \Big) \\ &= -\gamma \bar{y}_t^T \bar{P} \bar{y}_t + \dot{\bar{y}}_t^T \bar{P} \bar{y}_t + \bar{y}_t^T \bar{P} \dot{\bar{y}}_t \end{aligned} \tag{3-19}$$

Differentiating (3-5) results in

$$\dot{\bar{y}}_t = G\dot{x}(t) = GAx(t) + GBu(t)$$

Using this relation in (3-19) gives

$$\lim_{\Delta t \to 0} \frac{\bar{y}_t^T \bar{P} \bar{y}_t - e^{-\gamma \Delta t} \bar{y}_{t+\Delta t}^T \bar{P} \bar{y}_{t+\Delta t}}{\Delta t} = x(t)^T (A^T P + PA - \gamma P) x(t)$$

$$\lim_{\Delta t \to 0} \frac{\int_t^{t+\Delta t} e^{-\gamma(\tau - t)} \left[\bar{y}_t^T Q \bar{y}_t + u^T R u \right] d\tau}{\Delta t} + \lim_{\Delta t \to 0} \frac{\bar{y}_t^T \bar{P} \bar{y}_t - e^{-\gamma \Delta t} \bar{y}_{t+\Delta t}^T \bar{P} \bar{y}_{t+\Delta t}}{\Delta t}$$
$$= x(t)^T (A^T P + PA - \gamma P + C^T Q C) x(t) + u^T R u$$

Now, let G_2 be a filter with the same dimension of G, then

$$u^T R u = \hat{x}^T P B R^{-1} B^T P \hat{x} \tag{3-20}$$

where $\hat{x} \to x$ as $G_2 \to G$. The estimate G_2 can be made close to G by the adaptive algorithm as discussed before.
Hence the proof □

3-4 Discussion

- The advantage of such a novel algorithm is that it can be applied to systems where

  the systems matrices are uncertain.

  the system states are immeasurable.

  the first control update to be made is required to be faster.

- In the Algorithm-4, there are two processes that are combined to result in the PI algorithm. These processes are the parameter estimation process and the policy evaluation process. The parameter estimation process requires only an input which is persistently exciting whereas the input to the policy evaluation process is determined by K. So, the inputs to these processes do not affect each other. The algorithm as stated in Figure 3-3 is initialized with a zero input and a persistently exciting noise. This noise must be turned off ONLY after the system parameters converge. The noise also helps the policy evaluation process to converge. Since the noise is known beforehand, it can be considered into the IRL Bellman equation, to avoid affecting the convergence of the learning process.

- The parameter estimation process has $n+m+1$ parameters to be estimated (assuming B should also be estimated) and the policy evaluation process has $\frac{N(N+1)}{2}$ parameters to be identified. So, for the process to converge to a positive definite stable solution, the input noise discussed in the previous point should have $n+m+1$ or $\frac{N(N+1)}{2}$ distinct frequencies, whichever is higher.

- In the Algorithm-4, the parameters to be estimated also includes the elements of the B matrix. Since the B matrix is assumed to be known, these elements need not be estimated. This is a special case of the Algorithm-4, and therefore not stated explicitly.

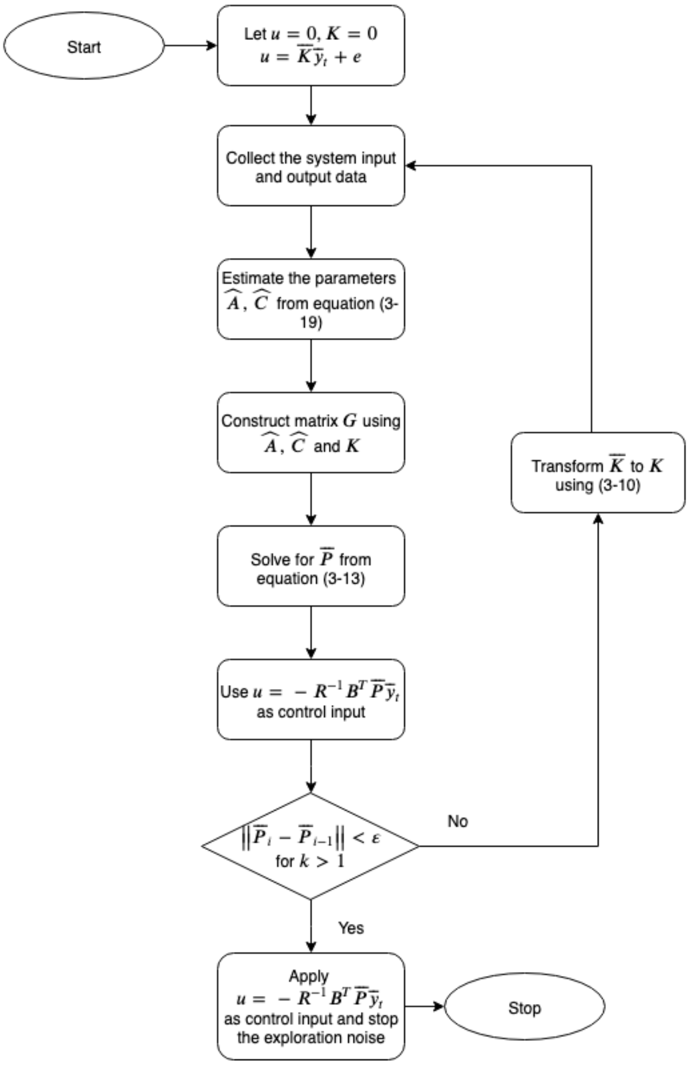

Start
Let $u = 0, K = 0$
$u = \overline{K}\,\overline{y}_t + e$
Collect the system input and output data
Estimate the parameters $\widehat{A}, \widehat{C}$ from equation (3-19)
Construct matrix G using $\widehat{A}, \widehat{C}$ and K
Solve for $\overline{P}$ from equation (3-13)
Use $u = -R^{-1}B^T\overline{P}\,\overline{y}_t$ as control input
$\left\|\overline{P}_i - \overline{P}_{i-1}\right\| < \varepsilon$ for $k > 1$
No
Yes
Transform $\overline{K}$ to K using (3-10)
Apply $u = -R^{-1}B^T\overline{P}\,\overline{y}_t$ as control input and stop the exploration noise
Stop

Figure 3-3: Flowchart for Algorithm-4

- The process is initialized with $K = 0$ as can be seen in (3-3). So, this algorithm can only work for systems where the system is known to be stable and the order of the system along with the relative degree is known.

- The algorithm as stated in Figure 3-3 can be used to solve tracking and a regulation problem. When the vector $\bar{y}_t$ is stacked with only the output data, the problem to be solved becomes a regulation problem. When the vector $\bar{y}_t$ is stacked with both the output data and the reference signal $r(t)$, the problem to be solved becomes a tracking problem. But there are conditions on the choice of γ for the type of problem. These conditions are the same as the ones stated in [31] and they are presented here briefly. The solution P is positive definite and one has

  $$Re(\lambda) < 0.5\gamma$$

  where λ is the eigenvalue of the closed-loop system A_c with

  $$A_c = A - BR^{-1}B^TP$$

  The closed-loop system is asymptotically stable if the condition is satisfied

  $$\gamma \leq \gamma^* = ||(BR^{-1/2})^T(Q^{1/2}C)^T||$$

- The solution for $\bar{P}$ in the policy evaluation step (3-13) is carried out in a Least Squares (LS) sense. In addition, (3-13) is a scalar equation and $\bar{P} \in N \times N$ is a symmetric matrix with $\frac{N(N+1)}{2}$ independent elements. Hence, at least $\frac{N(N+1)}{2}$ data points are required before (3-13) can be solved. Since $\frac{N(N+1)}{2}$ amount of data points are required before making one policy update, this process is not exactly online. A Recursive Least Squares (RLS) can be used to make this algorithm exactly online. When the RLS algorithm is used, one data point can be used to update both the policy evaluation step and the parameter estimation step.

Chapter 4

Thermostatically Controlled Load (TCL) Models Based on State Bins

Similar to the Chapter 3, another way to solve the problem (immeasurable states) is to use a model, where the state can be measured in real time. This chapter aims at discussing models where the state measurements are possible in reality and are also the state-of-the-art models.

4-1 Homogeneous model

Let us consider a homogeneous group of TCL where the primary model is represented by (2-1). The flux of loads moving within temperature bounds ($F(\cdot)$) can be represented by (4-1) where $X(\cdot)$ represents the load concentration at time t and Temperature T.

$$F_{on/off} = X_{on/off}\dot{\theta} \tag{4-1}$$

$\dot{\theta}$ is a function of ambient temperature and temperature set point. This relation is represented by (4-2). Here $\alpha(\cdot)$ represent the local diffusion rates.

$$F_{on/off} = \alpha_{on/off}(\theta_{amb}, \theta)X_{on/off} \tag{4-2}$$

The rate of increase of the load concentration is the difference between flux entering and exiting the control volume. This is represented by (4-3).

$$\frac{\partial X_{on/off}}{\partial t} = \frac{1}{dT}(F_{on/off}(T) - F_{on/off}(T + dT)) \tag{4-3}$$

Merging (4-2) and (4-3) results in (4-4). Here dT denotes the control volume length as can also be seen from Figure 4-1.

$$\frac{\partial X_{on/off}}{\partial t} = -\frac{\partial[\alpha_{on/off}X_{on/off}]}{\partial T} \tag{4-4}$$

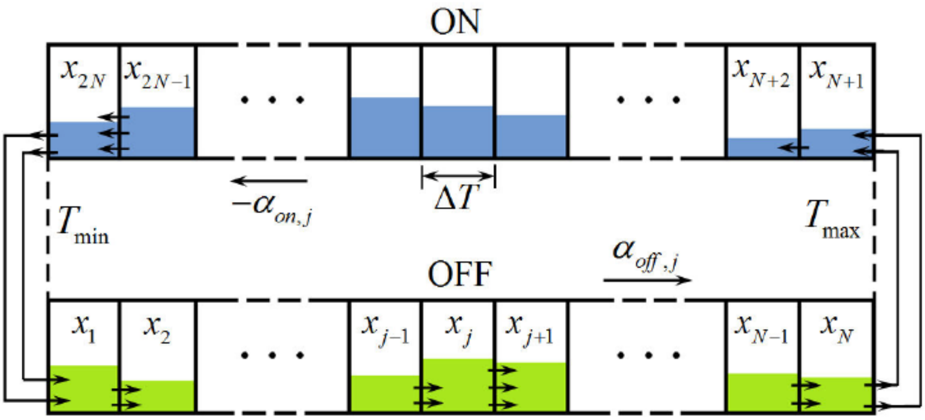


Figure 4-1: Discretized state bins [5]

(4-4) represents the free model where the set-point is constant. The forced model can be denoted by (4-5).

$$\frac{\partial X_{on/off}}{\partial t} = -\frac{\partial[(\alpha_{on/off} - \dot{T}_{sp})X_{on/off}]}{\partial T} \tag{4-5}$$

A state space equation can be formulated by applying backward difference method to (4-5) and is given by (4-6)-(4-7). Here $\bar{\alpha}$ represents the average cooling/heating rate, Δ represents dead-band and $\Delta T = \frac{\Delta}{N}$. Here N decides the smallest control volume possible for control. The pictorial representation of the diffusion process along the discretized state bins is represented in Figure 4-1.

$$\dot{x}_1(t) = -\frac{\bar{\alpha}_{off} - \dot{T}_{sp}}{\Delta T}x_1(t) - \frac{\bar{\alpha}_{on} - \dot{T}_{sp}}{\Delta T}x_{2N}(t) \tag{4-6}$$

$$\dot{x}_{N+1}(t) = -\frac{\bar{\alpha}_{off} - \dot{T}_{sp}}{\Delta T}x_N(t) - \frac{\bar{\alpha}_{on} - \dot{T}_{sp}}{\Delta T}x_{N+1}(t) \tag{4-7}$$

The above equations can be rearranged into a bi-linear state space model as follows [1]

$$\dot{x}(t) = Ax(t) + Bx(t)u(u), \quad u(t) = \dot{T}_{sp}(t) \tag{4-8}$$

$$y(t) = Cx(t), \tag{4-9}$$

[1] In this chapter, P represents the power drawn by the TCL.

where $A \in \mathbb{R}^{2N \times 2N}, B \in \mathbb{R}^{2N \times 2N}$ and $C \in \mathbb{R}^{2N \times 1}$ and

$$A = \begin{bmatrix} \frac{-\bar{\alpha}_{off}}{\Delta T} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{-\bar{\alpha}_{on}}{\Delta T} \\ \frac{\bar{\alpha}_{off}}{\Delta T} & \frac{-\bar{\alpha}_{off}}{\Delta T} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \frac{\bar{\alpha}_{off}}{\Delta T} & \frac{-\bar{\alpha}_{off}}{\Delta T} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\bar{\alpha}_{off}}{\Delta T} & \frac{\bar{\alpha}_{on}}{\Delta T} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{-\bar{\alpha}_{on}}{\Delta T} & \frac{\bar{\alpha}_{on}}{\Delta T} & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{-\bar{\alpha}_{on}}{\Delta T} & \frac{\bar{\alpha}_{on}}{\Delta T} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{-\bar{\alpha}_{on}}{\Delta T} & \frac{\bar{\alpha}_{on}}{\Delta T} \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{1}{\Delta T} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\Delta T} \\ \frac{-1}{\Delta T} & \frac{1}{\Delta T} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \frac{-1}{\Delta T} & \frac{1}{\Delta T} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{-1}{\Delta T} & \frac{-1}{\Delta T} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\Delta T} & \frac{-1}{\Delta T} & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\Delta T} & \frac{-1}{\Delta T} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\Delta T} & \frac{-1}{\Delta T} \end{bmatrix} \qquad C^T = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \frac{P}{\eta} \\ \vdots \\ \frac{P}{\eta} \end{bmatrix}$$

4-2 Heterogeneous model

Considering a heterogeneous group of TCLs that are initially completely ON/OFF. Then, the probability of the TCLs going from θ_{start} to θ_{end} is

$$P(\theta_{end}|\theta_{start}) = P(a_i)$$

where

$$a_i = \frac{\theta_a - \theta_{end} - m_t\theta_g}{\theta_a - \theta_{start} - m_t\theta_g}.$$

Similarly, the probability of the TCL going from $\theta_m < \theta_{start} < \theta_{m+1}$ to $\theta_n < \theta_{end} < \theta_{n+1}$ is

$$P(\theta_n < \theta_{end} < \theta_{n+1}|\theta_m < \theta_{start} < \theta_{m+1}) = \int_{\theta_m}^{\theta_{m+1}} \int_{a_1}^{a_2} p(a)\, da\, d\theta_{start} \tag{4-10}$$

where

$$a_1 = \frac{\theta_a - \theta_1 - m_t\theta_g}{\theta_a - \theta_{start} - m_t\theta_g}$$

$$a_2 = \frac{\theta_a - \theta_2 - m_t\theta_g}{\theta_a - \theta_{start} - m_t\theta_g}$$

where,

- $\theta_1 = \theta_{n/n+1}$ and $\theta_2 = \theta_{n+1/n}$ when the TCL is traversing from low/high to high/low temperature respectively.
- θ_g is the ON temperate gain of the TCL given by RP for cooling devices.
- m is a Boolean variable 1/0 defining the ON/OFF state of the TCL respectively.

Since this probability depends on the temperature gains, the parameter heterogeneity is inbuilt in this formulation. Let us divide the temperature dead-band of the TCL into several state bins as represented in Figure 4-1 and let the number of state bins be represented by N. When the equation (4-10) is evaluated for every starting and ending bins, the $A \in \mathbb{R}^{2N \times 2N}$ matrix can be analytically derived. The system matrix A can also be identified by considering it as an autonomous system [38]. Therefore in this model,

- The state $x \in \mathbb{R}^{2N}$ represent the number of TCL in each temperature bins.
- The control input $u \in \mathbb{R}^{N}$ represents the number of TCL to be switched from ON/OFF to OFF/ON respectively. The matrix B can be constructed as in (4-11).
- The output y represents the aggregate power of TCLs. The matrix C can be constructed as in (4-11).
- The state space representation of this model is represented in (4-12).

$$B = \begin{bmatrix} -1 & \dots & 0 \\ \vdots & \dots & \vdots \\ \vdots & \dots & -1 \\ 0 & \dots & 1 \\ \vdots & \dots & \vdots \\ 1 & \dots & 0 \end{bmatrix} \qquad C^T = P \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \tag{4-11}$$

$$\dot{x} = Ax + Bu \tag{4-12}$$

4-3 Discussion

The two models discussed in this chapter are (4-8) and (4-12). The key points about these models are

The model (4-8) is bi-linear and the model (4-12) is linear.

The state $x(.)$ in both cases represent the number of TCLs present in a particular temperature band (which can also be called as state bins) where the temperature lies within the dead-band of the TCL.

The sum of the state $\sum_n x(.)$ represents the total number of TCL in the group which is constant. Hence the individual states cannot be driven to driven simultaneously to 0. i.e. the controllability of the systems is $N - 1$.

The input $u(.)$ represents the derivative of the set-point to be applied to the system in the former and represents the number of TCLs to be switched prematurely from *ON/OFF* to *OFF/ON* state respectively in the latter. This difference in input causes the model to vary from a bi-linear to linear.

The output $y(.)$ is the aggregate power consumption of the TCL population.

Due to the constraints on the state (and the bi-linearity of the former model), the linear model algorithms (Algorithm-1 and Algorithm-2) cannot be applied to these systems.

The next section concentrates on applying the non-linear control algorithm ([6]) to the models developed in this section.

4-4 Non-linear control

A control algorithm for a non-linear affine system is discussed in this section. The block diagram of such a setting is represented in Figure 4-2. An online learning algorithm with semi-global stabilization for non-linear systems in the domain of attraction which can be made arbitrarily large is discussed in this section.

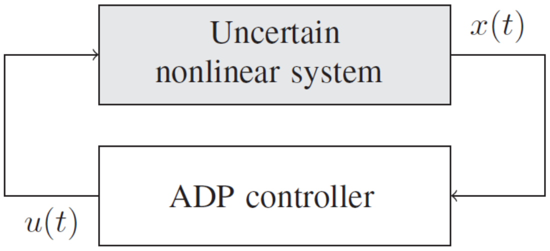


Figure 4-2: ADP-based online learning control for uncertain non-linear system [6]

4-4-1 Problem formulation

Consider a non-linear system of the form (4-13) where $x(.) \in \mathbb{R}^n$, $u(.) \in \mathbb{R}^m, f(.) : \mathbb{R}^n \to \mathbb{R}^n$ and $g(.) : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ and $f(.).g(.)$ are Lipschitz continuous functions. The objective is to find a control input $u(.)$ that minimizes the cost function (4-14). Here $r(.)$ can be defined as in (4-15).

$$\dot{x} = f(x) + g(x)u \tag{4-13}$$

$$J(x,u) = \int_0^\infty r(x(t),u(t))dt, \quad x(0) = x_0 \tag{4-14}$$

$$r(x(),u(.)) = q(x) + u^T R(x)u, \quad q(.) > 0, \quad R(.) > 0 \tag{4-15}$$

4-4-2 Non-linear off-policy optimal adaptive algorithm

Consider the system (4-13) which can be rewritten into the form (4-16). Here u_0 is the control input satisfying Assumption-2-$\mathcal{A}$. Due to the existence of exploration noise e, the Assumption-3-$\mathcal{A}$ should be satisfied. The system can be rewritten into the form (4-17) where $v_i = u_0 - u_i + e$.

$$\dot{x} = f(x) + g(x)(u_0 + e) \tag{4-16}$$

$$\dot{x} = f(x) + g(x)u_i(x) + g(x)v_i \tag{4-17}$$

Assumption 2-$\mathcal{A}$. *There exists a feedback control policy that globally stabilizes the system (4-13) with finite cost.*

Assumption 3-$\mathcal{A}$. *The system (4-16) is Input to State Stable (ISS)[39] when e is considered as input.*

The solution of the value function along the trajectory of (4-17) can be found by

$$\begin{aligned}\dot{V_i} &= \nabla V_i^T(x)[f(x) + g(x)u_i(x) + g(x)v_i] \\ &= -q(x) - u_i^T R(x)u_i - \nabla V_i^T(x)g(x)v_i \\ &= -q(x) - u_i^T R(x)u_i - 2u_{i+1}^T R(x)v_i \end{aligned} \tag{4-18}$$

Integrating (4-18) on the interval $[t, T+T]$ yields

$$V_i(x(t+T)) - V_i(x(t)) = -\int_t^{t+T} [q(x) + u_i^T R(x)u_i + 2u_{i+1}^T R(x)v_i]d\tau \tag{4-19}$$

By approximation theory [40], the value function and the control input ($V(.)$ and $u(.)$) can be approximated by basis function are represented by (4-20) and (4-21) where $\hat{c}$ and $\hat{w}$ are weights to be determined when N_1 and N_2 are sufficiently large. Using (4-20) and (4-21), (4-19) becomes (4-22). The solution $\hat{c}$ and $\hat{w}$ can be found by minimizing $e_{i.k}$ in a least squares sense. The equation (4-22) does not depend on the system dynamics but only on the state and input measurements. This brings us to the online adaptive Algorithm-5.

$$\hat{V_i}(x) = \sum_{j=1}^{N_1} \hat{c}_{i,j}\phi_j(x) \tag{4-20}$$

$$\hat{u}_{i+1}(x) = \sum_{j=1}^{N_2} \hat{w}_{i,j}\psi_j(x) \tag{4-21}$$

$$\begin{aligned}\sum_{j=1}^{N_1} \hat{c}_{i,j}[\phi_j(x(t_{k+1})) - \phi_j(x(t_k))] = -\int_{t_k}^{t_{k+1}} [q(x) + \hat{u}_i^T R(x)\hat{u}_i dt - \\ \int_{t_k}^{t_{k+1}} 2\sum_{j=1}^{N_2} \hat{w}_{i,j}\psi_j^T(x)R(x)\hat{v}_i dt + e_{i,k}\end{aligned} \tag{4-22}$$

Algorithm 5: Value Iteration (VI) algorithm for non-linear non-affine systems

```
Result: Weights of the basis functions $\hat{w}, \hat{c}$
1 Input: A initial stabilizing control policy

2 Initialization: Determine the set $\Omega \in \mathbb{R}^n$ for approximation. Find an initial control
  policy $u^0$ and $i \leftarrow 0$
3 Online data collection: Apply the initial control policy $u = u_0 + e$ and collect the
  system state and input information.
4 Policy evaluation and improvement: Solve for $\hat{w}$ and $\hat{c}$ from (4-22).
5 Stopping criterion: Let $i \leftarrow i + 1$, and go to Step 3, until

      $$\sum_{j=1}^{N_1} |\hat{c}_{i,j} - \hat{c}_{i-1,j}|^2 \leq \epsilon_1$$

  where $\epsilon_1 > 0$ is sufficiently small predefined threshold.
6 Actual control policy improvement: Terminate the exploration noise $e$ and
  $u = u_0$ as the control input. Once $x(t) \in \hat{\Omega}_i$, apply the control policy $u = \hat{u}_{i+1}$.
```

4-5 Results

The system (4-13) can be related to the model developed in section 4-2 and section 4-1 as $f(x)$ representing $A(x - x_{set})$ and $g(x)$ representing Bxu in the bi-linear case and Bx in the linear case. x_{set} represents the set-point power represented in terms of the distribution of TCL across state bins. The matrices A, B and C for the bi-linear model are constructed using the parameters represented in Table 2-1. The matrix A constructed by this method is shown in (4-23). The Algorithm-4 was applied to the bi-linear model and the output trajectory and the inputs are shown in Figure 4-3 whereas the norm convergence of $\phi(\cdot)$ and $\psi(\cdot)$ is shown in Figure 4-4. For the linear model, the parameters from Table 2-1 are used to simulate an uncontrolled diffusion process and the state transition matrix is identified. This matrix is shown in (4-24). The Algorithm-4 was applied to the linear model and the output trajectory and the inputs are shown in Figure 4-5 whereas the norm convergence of $\phi(\cdot)$ and $\psi(\cdot)$ is shown in Figure 4-6. The number of state bins considered in both these cases is 2.

$$A = \begin{bmatrix} -0.075 & 0 & 0 & 0.100 \\ 0.075 & -0.100 & 0 & 0 \\ 0 & 0.100 & -0.100 & 0 \\ 0 & 0 & 0.100 & -0.100 \end{bmatrix} \tag{4-23}$$

$$A = \begin{bmatrix} -0.620 & 0 & 0.760 & 0 \\ 0.620 & -0.570 & 0 & 0 \\ 0 & 0 & -0.760 & 0.830 \\ 0 & 0.570 & 0 & -0.820 \end{bmatrix} \tag{4-24}$$

Remark: Since the system involves state coordinate transformation, it is necessary to have an approximate knowledge about the system dynamic matrix (A). The objective of this thesis is to study the effect of parameter heterogeneity, the model developed in [18] is considered more relevant. A detailed discussion about its advantages and tracking performance is discussed in the next chapter.

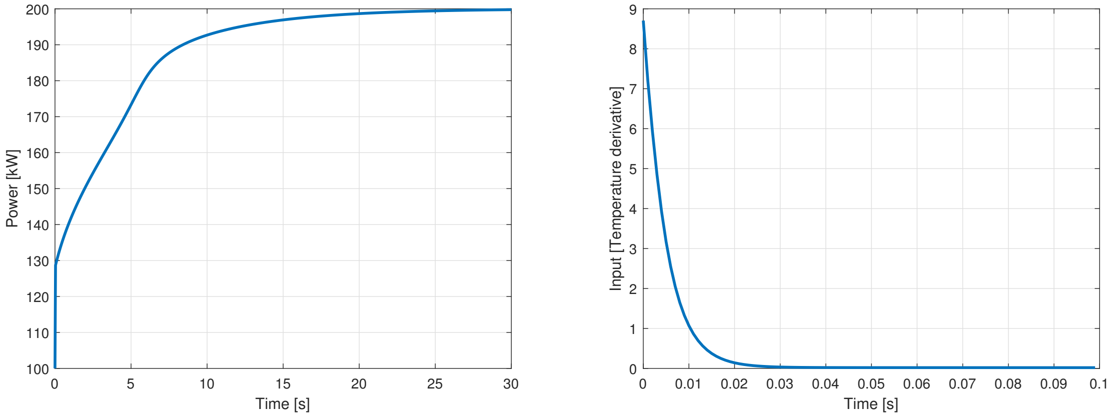


Figure 4-3: Output trajectory and control input, bi-linear model

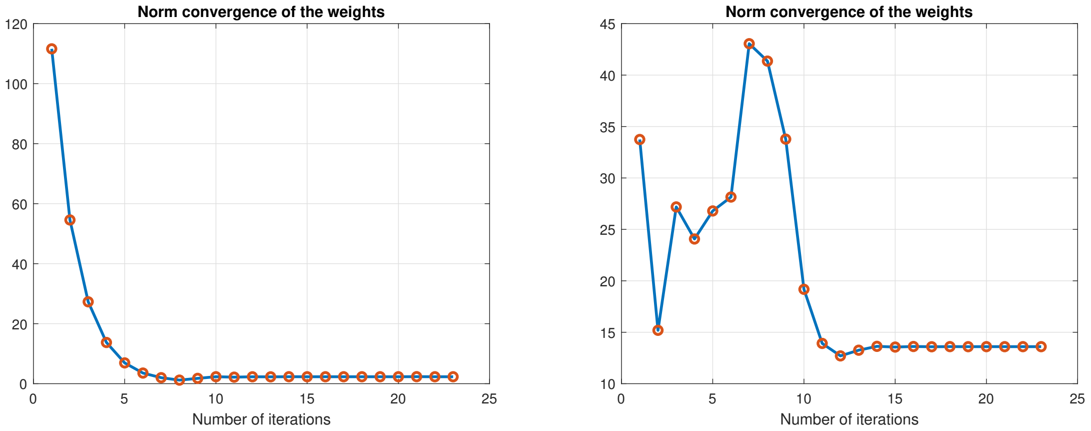


Figure 4-4: Norm convergence of weights, bi-linear model

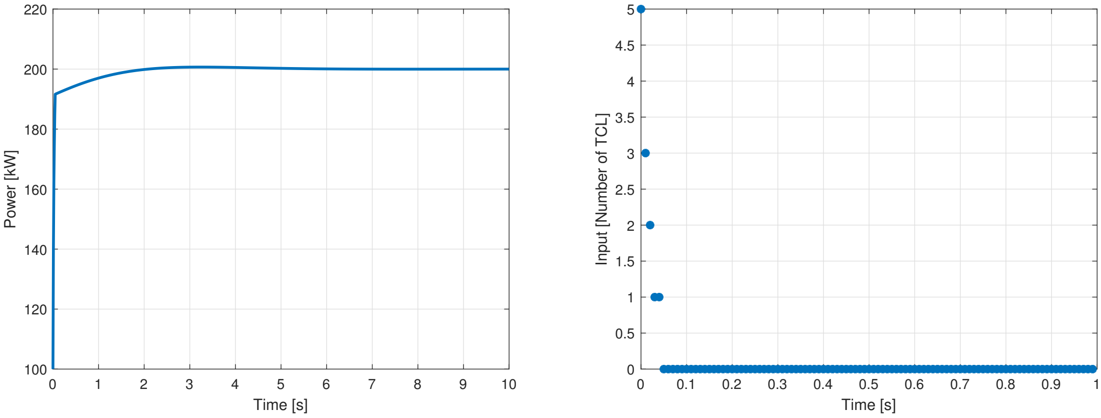


Figure 4-5: Output trajectory and control input, linear model

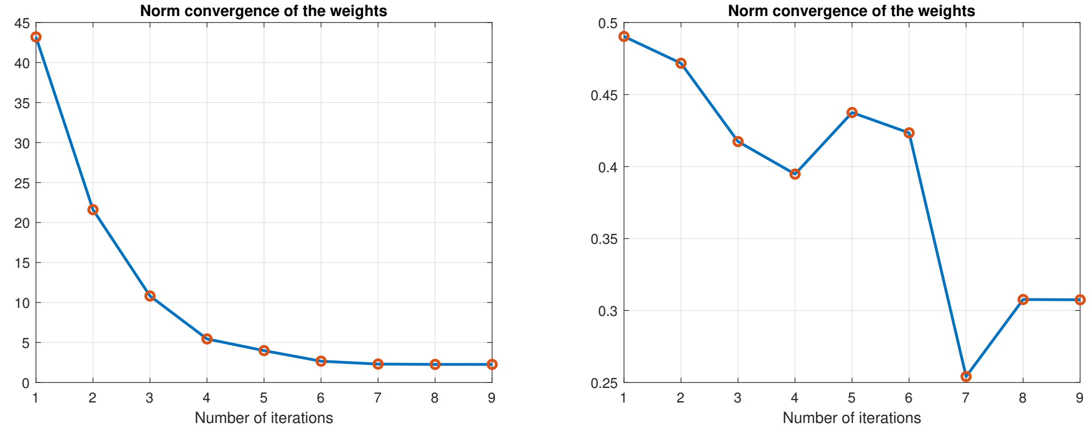


Figure 4-6: Norm convergence of weights, linear model

Chapter 5

Results and Discussion

This chapter aims at answering the Research Question (RQ) 3. From the models discussed in Chapter 4, it is interesting to note that the model discussed in section 4-2 is the most appropriate model for the following reasons.

- This model can be used to represent both a homogeneous and heterogeneous population.
- This model represents a system where the states are completely measurable.
- The control input is practically implementable since it is easier to switch the state of the TCL than changing the derivative of the set-point as in the model discussed in section 4-1.
- The system can also be easily identified easily as discussed in the Appendix A since the states are measurable.

This model is also widely used in literature as in [18] and references therein. This chapter is dedicated to exploring the model more in detail and to study the effect of the non-linear control approach as discussed in section 4-4-2 to this model. For a more structured approach, this chapter is divided in such a way to answer the following questions.

- How does the varying set-points affect the performance?
- How does the varying input weights (R) affect the performance?
- How does this model perform in comparison to the second-order model (2-8)-(2-9)?
- How does the heterogeneity affect the performance?
- How does the number of state bins affect the performance?

5-1 Effect of varying set-point

Consider a heterogeneous population of TCLs where

- The number of state bins considered is 4.
- The number of TCLs considered is 80.
- The input is considered to be synchronous. That is: consider a state model where the number of bins is N. The number of control input becomes N. The input applied to all these inputs is assumed to be the same.
- During the learning process, a noise of the form $u = \sum_{\omega=1}^{30} sin(\omega t)$ is applied to the system to persistently excite the system.
- The parameter heterogeneity is assumed to occur only the system resistance (R) and capacitance (C). But the power drawn by the TCL is assumed to be completely known and homogeneous in distribution.

As mentioned in section 4-4-2, basis functions or function approximations is necessary to approximate $\phi(\cdot)$ and $\psi(\cdot)$. In this particular case, the function approximates are defined as follows

$$x_1^2,\ x_2^2,\ x_3^2,\ x_4^2,\ x_5^2,\ x_6^2,\ x_7^2,\ x_8^2,\ x_1x_2,\ x_1x_3,\ x_1x_4,\ x_2x_3,\ x_2x_4,\ x_3x_4,$$
$$x_1^4,\ x_2^4,\ x_3^4,\ x_4^4,\ x_5^4,\ x_6^4,\ x_7^4,\ x_8^4,\ x_1^2x_2^2,\ x_1^2x_3^2,\ x_1^2x_4^2,\ x_2^2x_3^2,\ x_2^2x_4^2,\ x_3^2x_4^2$$

$$x_1,\ x_2,\ x_3,\ x_4,\ x_5,\ x_6,\ x_7,\ x_8,\ x_1x_1x_1,\ x_1x_1x_2,\ x_1x_1x_3,\ x_1x_1x_4,$$
$$x_1x_2x_2,\ x_1x_2x_3,\ x_1x_2x_4,\ x_1x_3x_3,\ x_1x_3x_4,\ x_1x_4x_4,\ x_2x_2x_2,\ x_2x_2x_3,$$
$$x_2x_2x_4,\ x_2x_3x_3,\ x_2x_3x_4,\ x_2x_4x_4,\ x_3x_3x_3,\ x_3x_3x_4,\ x_3x_4x_4,\ x_4x_4x_4$$

Here x_i represent the $i-th$ state of the system. Here a heterogeneous population of TCL with the parameters mean $R = 2, C = 10$ and distributed with a variance of 0.5 is used to simulate the temperature dynamics. The temperature response of such a system is shown in Figure 5-1.

The number of TCLs in a given state bin is collected over a period of 50 seconds and the response (partly) is shown in Figure 5-2. This data is used to identify a system where the system dynamics are given by (5-1). It is important to note that $\sum_{i=1}^{8} x_i$ is the total number of TCLs.

$$A = \begin{bmatrix} -1.3849 & 0.0080 & 0.0211 & -0.0403 & 1.6074 & 0.1744 & -0.0814 & -0.0547 \\ 1.3686 & -1.3777 & 0.1377 & -0.3212 & 0.2474 & -0.1431 & -0.1505 & 0.3851 \\ -0.0160 & 1.4752 & -1.2884 & 0.3092 & -0.0001 & -0.3320 & 0.2819 & -0.5537 \\ -0.1468 & 0.0811 & 1.2804 & -1.1789 & -0.2485 & 0.2073 & -0.1386 & 0.2382 \\ 0.0517 & 0.1863 & -0.3284 & 0.2158 & -1.6444 & 1.5340 & 0.2140 & -0.1963 \\ 0.2135 & -0.3035 & 0.2025 & -0.0632 & 0.0495 & -1.6396 & 1.6664 & -0.0845 \\ -0.2896 & 0.1165 & -0.0454 & -0.0683 & 0.0728 & 0.1680 & -1.7430 & 1.9211 \\ 0.2036 & -0.1858 & 0.0204 & 1.1469 & -0.0842 & 0.0310 & -0.0488 & -1.6552 \end{bmatrix} \tag{5-1}$$

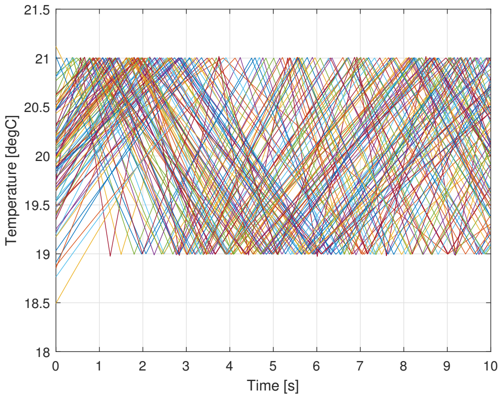


Figure 5-1: Temperature dynamics

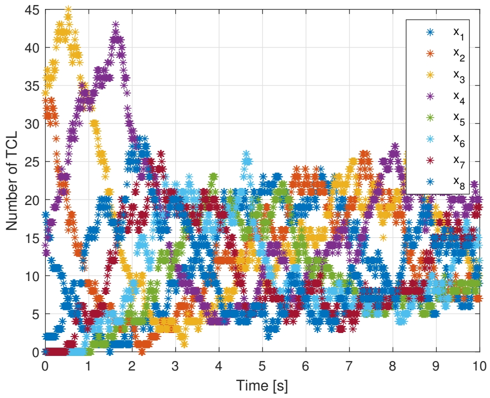


Figure 5-2: Bin dynamics

The learning algorithm is performed with this data and the resulting weights of the basis function of ψ are shown in Table 5-1.

The learning algorithm is performed on a system that is transferred in co-ordinates. Consider a system which is identified by subspace identification as mentioned in Appendix A and let the system dynamics be represented by

$$\dot{x} = Ax.$$

Since the power drawn by the individual TCLs is known, the matrix C can be explicitly constructed which means that the reference in terms of power can be transferred into the number of TCLs to be ON/OFF (the system states - x_{set}). Hence the system can be transferred to

$$\dot{x} = A\bar{x} \quad \text{where} \quad \bar{x} = x - x_{set}.$$

The reference set-point is a parameter in the learning algorithm. The above weights are learned for a system for which the set-point is $\begin{bmatrix}0 & 0 & 0 & 0 & 20 & 20 & 20 & 20\end{bmatrix}^T$. The output matrix C used in this study is $C = \begin{bmatrix}0 & 0 & 0 & 0 & 5 & 5 & 5 & 5\end{bmatrix}$. Let us say, the power to be tracked is 300 kW and 400 kW. This can be transferred to the states as $\begin{bmatrix}5 & 5 & 5 & 5 & 15 & 15 & 15 & 15\end{bmatrix}^T$

x_1	0.4573	x_2	-1.1192
x_3	0.3883	x_4	-0.3147
x_5	0.7206	x_6	-0.1558
x_7	-0.1717	x_8	0
$x_1x_1x_1$	-0.6770	$x_1x_1x_2$	3.7618
$x_1x_1x_3$	-3.1089	$x_1x_1x_4$	1.3090
$x_1x_2x_2$	-6.9359	$x_1x_2x_3$	11.4138
$x_1x_2x_4$	-4.8117	$x_1x_3x_3$	-4.6663
$x_1x_3x_4$	3.9254	$x_1x_4x_4$	-0.8290
$x_2x_2x_2$	4.2568	$x_2x_2x_3$	-10.4759
$x_2x_2x_4$	4.4279	$x_2x_3x_3$	8.5442
$x_2x_3x_4$	-7.2024	$x_2x_4x_4$	1.5215
$x_3x_3x_3$	-2.3111	$x_3x_3x_4$	2.9142
$x_3x_4x_4$	-1.2266	$x_4x_4x_4$	0.1718

Table 5-1: Weights of function approximates

and $\begin{bmatrix}15 & 15 & 15 & 15 & 5 & 5 & 5 & 5\end{bmatrix}^T$. It is to note that this transformation is not unique, and a solution is valid when it follows the below mentioned rules:

- The sum of the total number of TCLs should result in a positive integer while the individual elements also remain positive.
- The number of TCL in any state bin cannot be zero. It is because the TCLs traverse around the dead-band in a counter-clockwise direction as represented in Figure 4-1. Making the reference zero in one state-bin means that the TCL has to traverse without entering into a particular state bin which is impossible.
- An exception to the previous case can be when all the TCLs are completely ON/OFF.

The weights (Table 5-1) is used to apply input to these reference points and the simulated results are shown in Figure 5-3 to Figure 5-5. The following inferences can be made from the plots

- The initial state of the system is equidistant from 100 kW and 300 kW and hence has the same inputs.
- Since the references are in the opposite direction (increase and decrease of power from the initial condition), the inputs are negated as expected.
- Also, the references with higher magnitude have higher control inputs. It is possible in real life scenarios, that this magnitude of the control input is limited. Hence a study of varying input weights R is studied next.

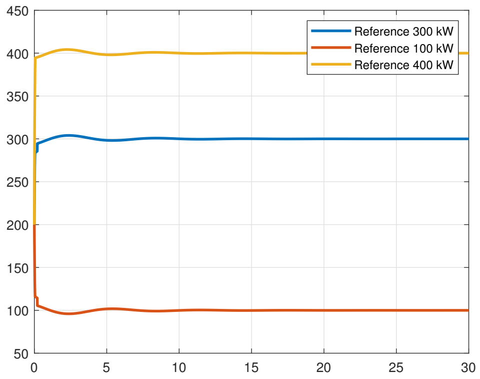


Figure 5-3: Output trajectories for different set-points

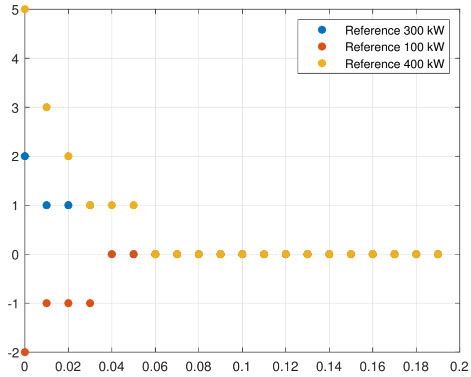


Figure 5-4: Control inputs for different set-points

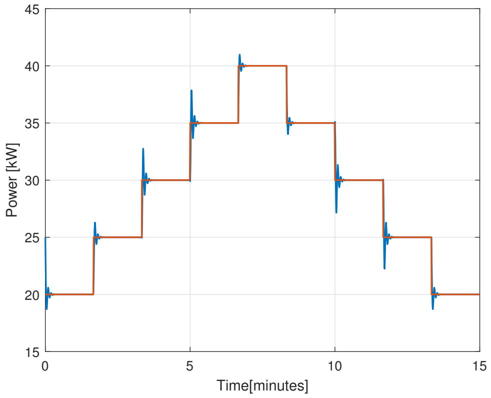


Figure 5-5: Output trajectory for varying set-points

5-2 Effect of varying input weights

The input weight (R) plays an important role because

- There can a physical limitation in the number of TCLs that can be switched but can deal with longer settling time.
- Although the states are physically represented by the system, in simulations, there can be situations where the states do not physically mean anything. For example, a state with negative values has no physical interpretation.

The same system matrices as discussed in (5-1) is used to simulate the following results (Figure 5-6 and Figure 5-7) but with a different input weight. As the input weight R increases, the magnitude of the input decreases, the settling time increases, and the cost increases. The increase of the cost might not be evident from the plots but the cost is calculated by using the equation $\sum_{t=0}^{t_n} y(t)^T y(t) + u(t)^T Ru(t)$ and is represented in the Table 5-2. The same objective of tracking can be achieved both by increasing the input weight R or by reducing the weight on Q.

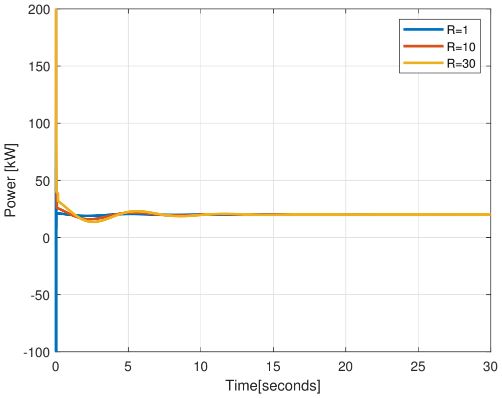


Figure 5-6: Output trajectory for varying input weights R

R	cost
1	125 10^3
10	127 10^3
30	128 10^3

Table 5-2: Cost for varying input weights R

5-3 Comparison to second-order model

The system represented in (2-8)-(2-9) is a second-order model of a homogeneous population. A similar setting is created here to track 20 kW and the results are shown in Figure 5-8. The

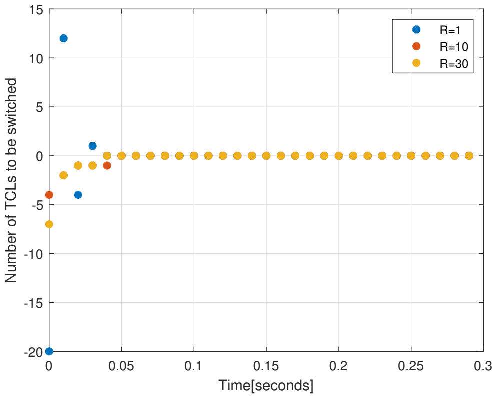


Figure 5-7: Control input for varying input weights R

norm convergence of the system weights during the learning process is shown in Figure 5-9. The system matrix (a homogeneous population) is shown in (5-2). From this system matrix it can be noted that, for a homogeneous population as in this case, most of the elements are zero except the diagonal elements and its adjacent elements. This means that there is a transition from the current state bin to the next state-bin or its same state which also happens in real life as mentioned in Figure 4-1. There are some critical differences between the models compared in this section.

- The former has an input that represents the deviation of the set-point temperature whereas in the latter case, the input represents the number of TCL to be switched.
- The latter has a more desirable situation because the same objective is achieved by making the TCLs stay within the initially defined dead-band.
- Since the A matrix is identified rather than derived, the latter would be more robust.
- In reality, since the power drawn by the TCL in ON state is known and OFF state is assumed to be zero, it is very relatable as to the effect of input on the output in contrast to the former model.
- In the latter model, a faster convergence is practical/a reality.

$$A = \begin{bmatrix} -1.2903 & 0.0000 & -0.0000 & 0.0000 & 1.5366 & -0.0118 & 0.0001 & -0.0000 \\ 1.2983 & -1.2270 & 0.0000 & -0.0000 & -0.0100 & 0.0001 & -0.0000 & 0.0000 \\ -0.0080 & 1.2343 & -1.1834 & 0.0000 & 0.0001 & -0.0000 & 0.0000 & -0.0000 \\ 0.0001 & -0.0073 & 1.1901 & -1.1173 & -0.0000 & 0.0000 & -0.0000 & 0.0000 \\ 0.0000 & -0.0000 & 0.0000 & -0.0000 & -1.5267 & 1.5414 & -0.0128 & 0.0001 \\ -0.0000 & 0.0000 & -0.0000 & 0.0001 & 0.0000 & -1.5297 & 1.6656 & -0.0141 \\ 0.0000 & -0.0000 & 0.0001 & -0.0095 & -0.0000 & 0.0000 & -1.6529 & 1.6947 \\ -0.0000 & 0.0001 & -0.0067 & 1.1268 & 0.0000 & -0.0000 & 0.0000 & -1.6807 \end{bmatrix} \tag{5-2}$$

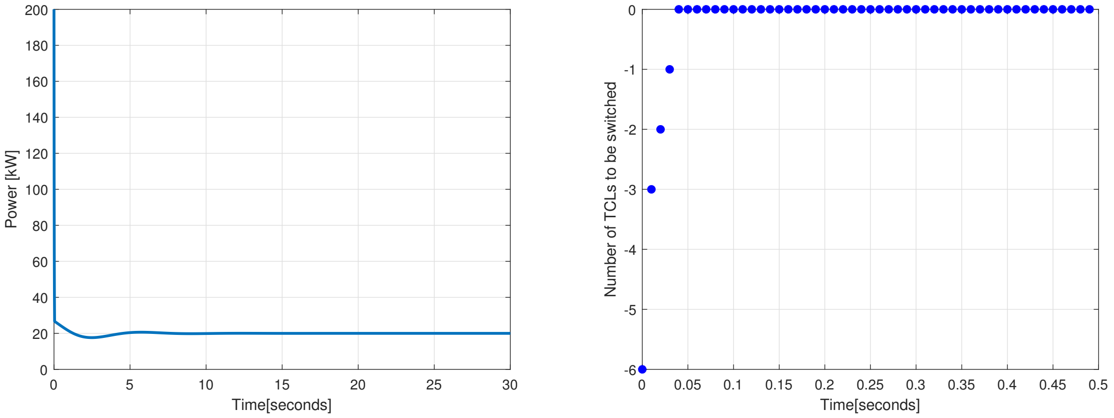


Figure 5-8: Output trajectory and control input for 20kW, homogeneous case

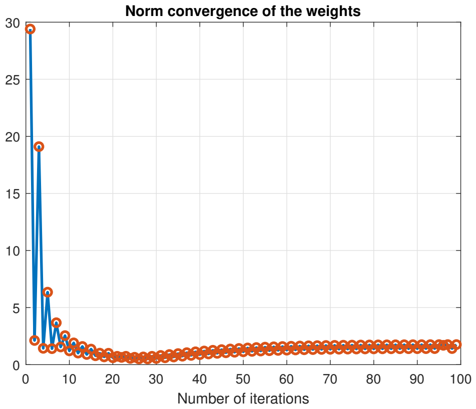


Figure 5-9: Norm convergence of weights

5-4 Effect of heterogeneity

A more detailed study on the effect of parameter heterogeneity on the tracking performance is conducted in this section. Let us consider the coordinate transferred system representation

$$\dot{x} = A\bar{x} \quad \text{where} \quad \bar{x} = x - x_{set}.$$

The controllable system can be represented as

$$\dot{x} = Ax - Ax_{set} + Bu.$$

Here, A is identified as mentioned in Appendix A. Now there are two questions which seems interesting which are

1. If the input weights $\psi(\cdot)$ are learnt for one particular system representation, can the same weights be applied for a system with a different parameter heterogeneity distribution?
2. Let the system be represented by $\dot{x} = A_1x - A_2x_{set} + Bu$. If the input weights $\psi(\cdot)$ are learnt for one particular system representation where A_1 and A_2 are identical, can the same weights be applied to for a system where A_1 and A_2 are different?

A system with 4 state bins is considered and the wights $(\psi(\cdot))$ are learned for a homogeneous population. These weights are applied to a system with different parameter heterogeneity and the results are shown in Figure 5-10. The simulation is carried out with $Q = 1, R = 5$. There is a difference in the rate of convergence, but the cost differs significantly. The cost is calculated by $\sum_{t=0}^{t_n} y(t)^T y(t)$ and the result is shown in Table 5-3. To provide a similar setting as of the second-order model (section 3-2) for comparison, a heterogeneous population with $\sigma = 0.1$ is made to track a reference of 20 kW and the result is shown in Figure 5-11 with the same weights R and Q. It is evident from the figure that the response is different significantly and the cost increases. This answers question 1. An experiment is conducted to determine the answer to question 2 in the following way: The weights of $\psi(\cdot)$ are learned for a system where $A_1 = A_2$ and the weights are applied to the system where $A_1 \neq A_2$. It is concluded that the system becomes unstable except for extremely low norm different matrices. Hence this algorithm works only when the system knowledge is known approximately.

σ	cost
0	$206\ 10^3$
0.1	$205\ 10^3$
0.2	$204\ 10^3$
0.4	$204\ 10^3$
0.6	$204\ 10^3$

Table 5-3: Cost for varying parameter heterogeneity

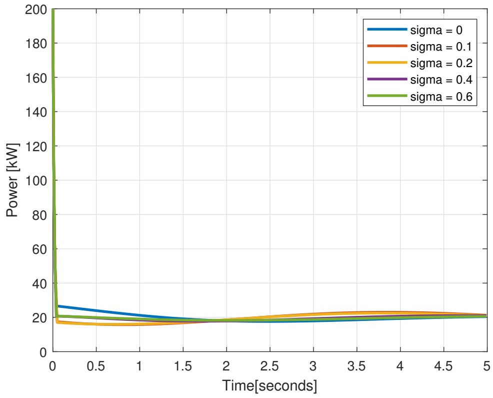


Figure 5-10: Output trajectories for varying parameter heterogeneity

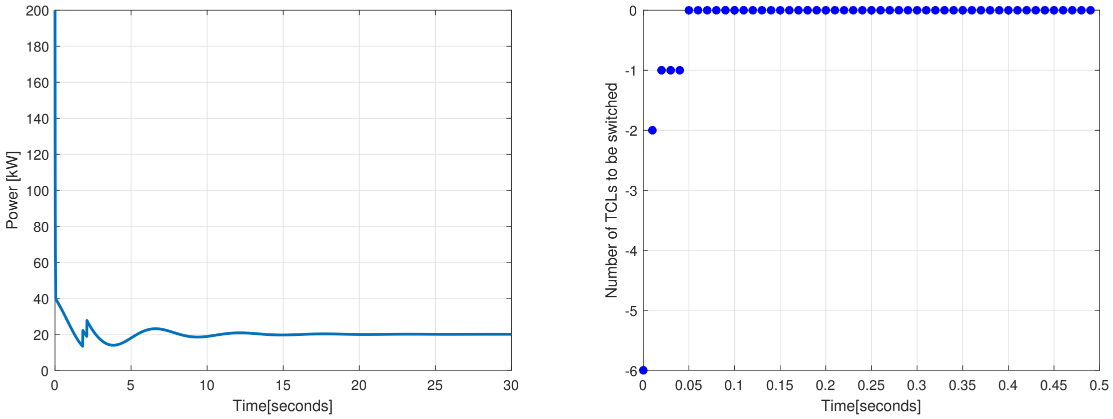


Figure 5-11: Output trajectory and control input for 20kW, heterogeneous case

5-5 Effect of the number of bins

This study is performed to study how the increase in the number of bins affects performance. The main advantage of using a system with a higher number of state bins is that the control becomes more precise. That is: the dead-band becomes discretized into smaller bins which makes it possible to accurately place certain TCLs in an accurate temperature band inside the dead-band. Consider a homogeneous ((5-3)-(5-5)) and a heterogeneous population ((5-6)-(5-8)) of TCLs with increasing number of state bins.

$$A = \begin{bmatrix} -0.62 & 0.00 & 0.76 & -0.00 \\ 0.62 & -0.57 & -0.00 & 0.00 \\ 0.00 & -0.00 & -0.76 & 0.83 \\ -0.00 & 0.57 & 0.00 & -0.82 \end{bmatrix} \tag{5-3}$$

$$A = \begin{bmatrix} -1.06 & 0.00 & -0.00 & 1.28 & -0.00 & 0.00 \\ 1.07 & -0.92 & 0.00 & -0.00 & 0.00 & -0.00 \\ -0.00 & 0.92 & -0.80 & 0.00 & -0.00 & 0.00 \\ 0.00 & -0.00 & 0.00 & -1.27 & 1.07 & -0.00 \\ -0.00 & 0.00 & -0.00 & 0.00 & -1.06 & 1.20 \\ 0.00 & -0.00 & 0.81 & -0.00 & 0.00 & -1.19 \end{bmatrix} \tag{5-4}$$

$$A = \begin{bmatrix} -1.62 & 0.00 & -0.00 & 0.00 & -0.00 & 1.92 & -0.01 & 0.00 & -0.00 & 0.00 \\ 1.63 & -1.55 & -0.00 & -0.00 & 0.00 & -0.01 & 0.00 & -0.00 & 0.00 & -0.00 \\ -0.01 & 1.56 & -1.60 & -0.00 & -0.00 & 0.00 & -0.00 & 0.00 & -0.00 & 0.00 \\ 0.00 & -0.01 & 1.61 & -1.46 & -0.00 & -0.00 & 0.00 & -0.00 & 0.00 & -0.00 \\ -0.00 & 0.00 & -0.01 & 1.47 & -1.38 & 0.00 & -0.00 & 0.00 & -0.00 & 0.00 \\ 0.00 & -0.00 & 0.00 & -0.00 & 0.00 & -1.90 & 1.96 & -0.01 & 0.00 & -0.00 \\ -0.00 & 0.00 & -0.00 & 0.00 & -0.00 & -0.00 & -1.94 & 1.85 & -0.01 & 0.00 \\ 0.00 & -0.00 & 0.00 & -0.00 & 0.00 & -0.00 & -0.00 & -1.83 & 2.10 & -0.02 \\ -0.00 & 0.00 & -0.00 & 0.00 & -0.01 & 0.00 & -0.00 & 0.00 & -2.08 & 2.14 \\ 0.00 & -0.00 & 0.00 & -0.01 & 1.40 & -0.00 & 0.00 & -0.00 & 0.00 & -2.12 \end{bmatrix} \tag{5-5}$$

$$A = \begin{bmatrix} -0.58 & -0.19 & 0.94 & 0.02 \\ 0.80 & -0.39 & -0.27 & -0.20 \\ -0.13 & 0.02 & -0.60 & 0.78 \\ -0.09 & 0.56 & -0.07 & -0.60 \end{bmatrix} \tag{5-6}$$

$$A = \begin{bmatrix} 1.11 & -0.19 & 0.04 & 1.53 & 0.03 & -0.05 \\ 1.45 & -1.09 & -0.06 & 0.11 & -0.18 & 0.03 \\ -0.16 & 1.23 & -0.74 & -0.10 & -0.23 & -0.05 \\ -0.15 & 0.24 & -0.16 & -1.49 & 1.43 & 0.01 \\ 0.10 & -0.19 & 0.01 & 0.22 & -1.44 & 1.41 \\ -0.12 & -0.00 & 0.91 & -0.27 & 0.38 & -1.35 \end{bmatrix} \tag{5-7}$$

$$A = \begin{bmatrix} -1.79 & -0.31 & 0.44 & -0.34 & 0.12 & 2.13 & 0.00 & 0.33 & -0.30 & 0.00 \\ 1.93 & -1.35 & -0.30 & 0.07 & -0.18 & -0.20 & 0.06 & -0.50 & 0.55 & 0.02 \\ -0.25 & 1.33 & -1.20 & -0.04 & 0.19 & 0.29 & -0.05 & 0.51 & -0.48 & -0.35 \\ 0.13 & 0.39 & 1.30 & -1.38 & -0.36 & -0.16 & -0.08 & -0.44 & 0.06 & 0.69 \\ -0.29 & -0.05 & 0.10 & 1.49 & -1.12 & 0.02 & 0.34 & -0.05 & -0.33 & -0.12 \\ 0.17 & 0.17 & -0.49 & 0.26 & 0.12 & -2.19 & 2.12 & 0.10 & -0.05 & -0.23 \\ -0.01 & -0.22 & 0.40 & -0.06 & -0.32 & 0.23 & -2.36 & 2.03 & 0.36 & 0.12 \\ -0.28 & 0.35 & -0.26 & -0.00 & 0.10 & 0.10 & 0.22 & -2.27 & 2.12 & -0.02 \\ 0.08 & -0.09 & -0.17 & 0.04 & 0.13 & 0.05 & 0.03 & 0.11 & -2.46 & 2.31 \\ 0.31 & -0.21 & 0.20 & -0.03 & 1.30 & -0.28 & -0.29 & 0.16 & 0.54 & -2.42 \end{bmatrix} \tag{5-8}$$

To understand the dynamics better, a comparison is made between the eigenvalues of a homogeneous and a heterogeneous population. As the number of state bins increases, the system becomes well represented and the area covered by the fictions circle connecting the eigenvalues increases. This can be well understood from Figure 5-12. This pattern changes when heterogeneity is introduced as can be more evidently seen in Figure 5-13. Although these matrices are identified, their validity has been proven in comparison to the analytically derived matrices in [18].

Considering this system $\dot{x} = Ax$ where the system is initialized at $x_i = 10\ \forall i$. The free response of the system is shown in Figure 5-14 and Figure 5-15. The system is stable as can be seen from the figures and controllable. It can be inferred from the figure that as the number of state bins increases, the number of oscillations and the settling time increases as well. It is possibly because of the more accurate system representation.

When the system is transformed as $\dot{x} = Ax - Ax_{set}$, the response changes as represented in Figure 5-16 and Figure 5-17. This plot represents $x - x_{set}$ vs time. The system settles to the reference point after a particular time. One might argue that, since the error dynamics are stable, a control input might not be necessary. But control input is necessary because of the reason that in certain cases a very fast convergence is preferred. Besides, the system dynamics represented in Figure 5-2 is for one particular case where the transferred system settles as can be seen from Figure 5-16 and Figure 5-17. But there might be cases where the system settles after a longer period in terms of hours. This simulation represents a possible direction of control methodology which can be extended to a system with different dynamics to have a desirable performance.

The non-linear control approach is applied for these systems for varying state bins and the results are shown in Figure 5-18 - Figure 5-21. These simulations are carried out with $Q = 1$ and $R = 5$. It is important to see that the magnitude of the control input increases as the number of bins increases, but this increase is due to different initial conditions as can be seen from the figures. In Figure 5-19, the control input is applied for a longer period in contrast to Figure 5-18 where a control input with a higher magnitude is applied for a shorter period. Although a system with a higher dimension makes the representation accurate, there is a compromise that needs to be made in terms of the computational effort as can be seen in Table 5-4. This time only represents the time taken by the algorithm to converge to a stable solution. But the algorithm is off policy and requires the system states before initializing the algorithm. So, the time to collect the states in addition to the time mentioned in Table 5-4 is the total time before which the policy can be updated. In general, there is no thumb rule to determine the number of data points necessary to guarantee a stable solution.

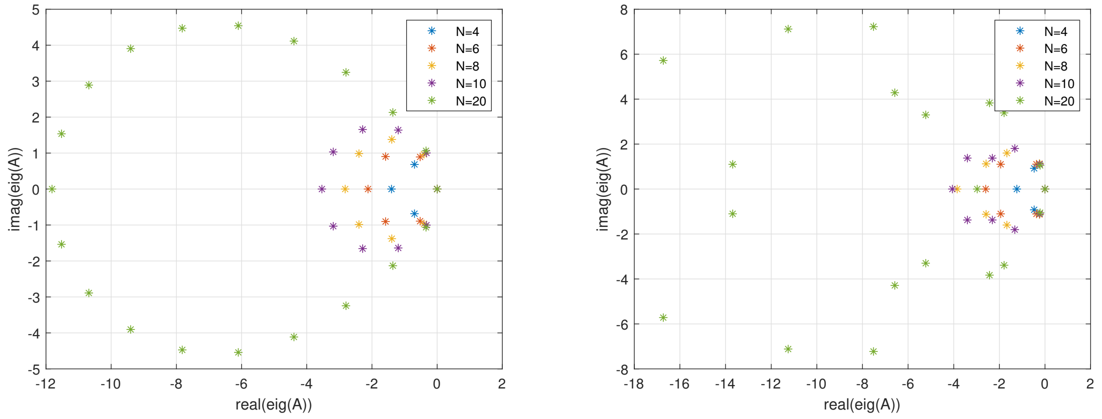


Figure 5-12: Eigenvalues, homogeneous and heterogeneous population

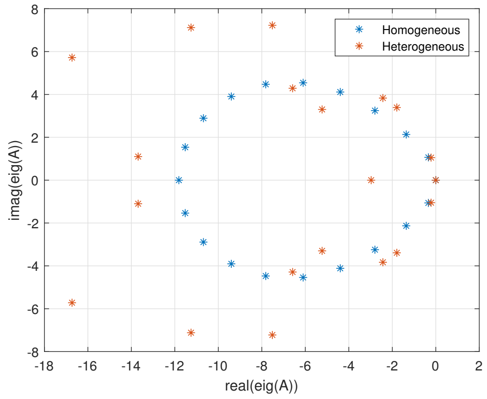


Figure 5-13: Eigenvalues comparison, N_{bin}=10

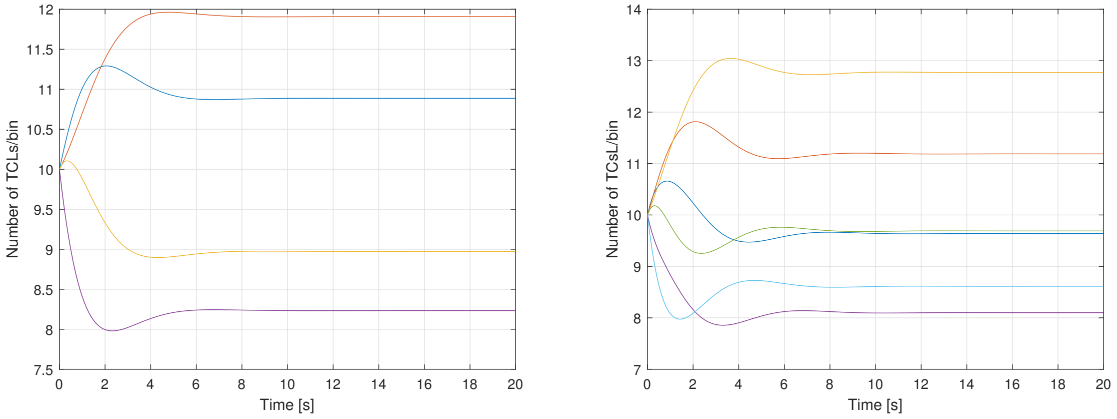


Figure 5-14: Free response for N_{bin} 2 and 3

Remark: The switching of TCLs that are represented in these plots denoted as inputs (which represents the number of TCLs to be switched) only represents the external switching. There

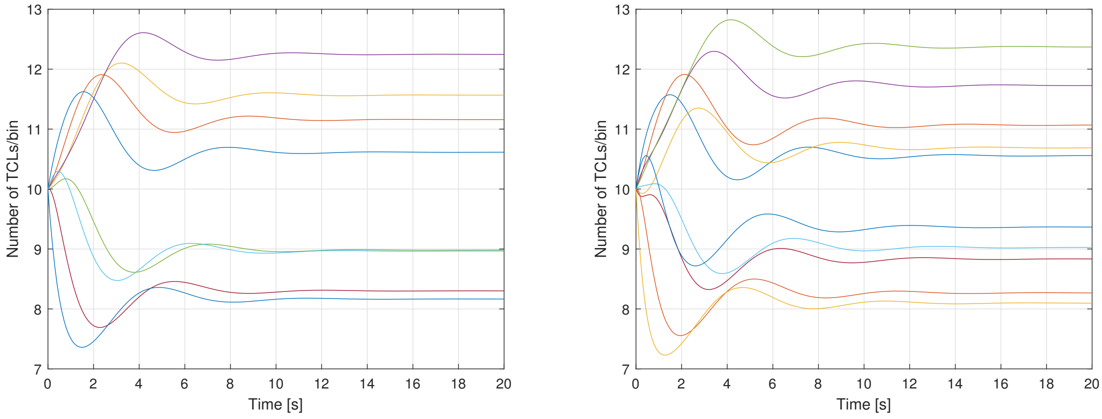


Figure 5-15: Free response for N_{bin} 4 and 5

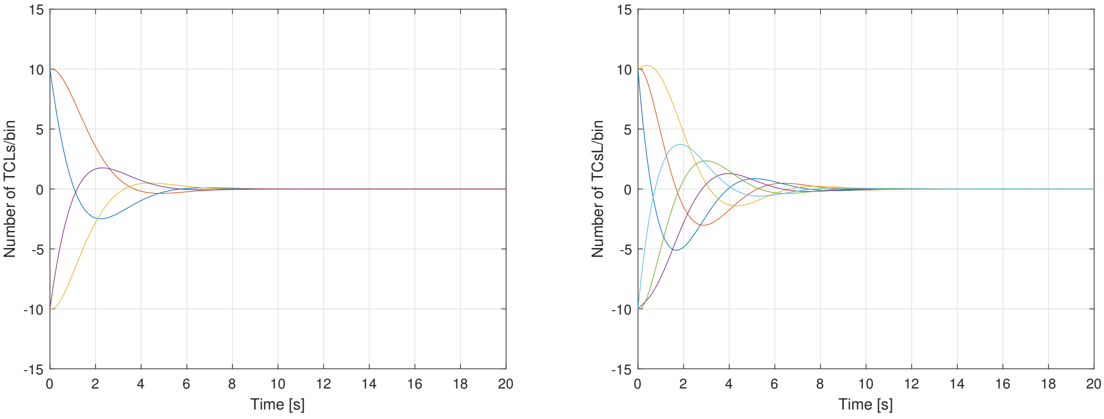


Figure 5-16: Free response for N_{bin} 2 and 3, transferred co-ordinates

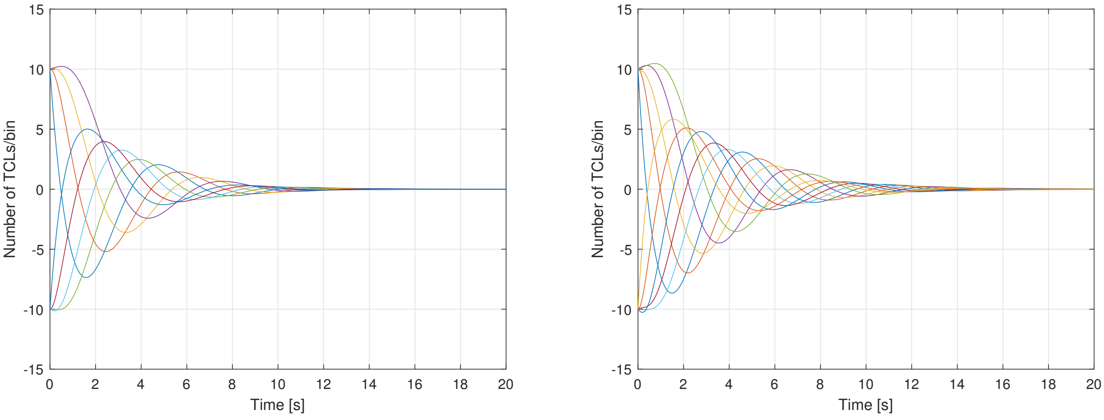


Figure 5-17: Free response for N_{bin} 4 and 5, transferred co-ordinates

is always an internal switching which keeps happening due to the internal dynamics of the TCL.

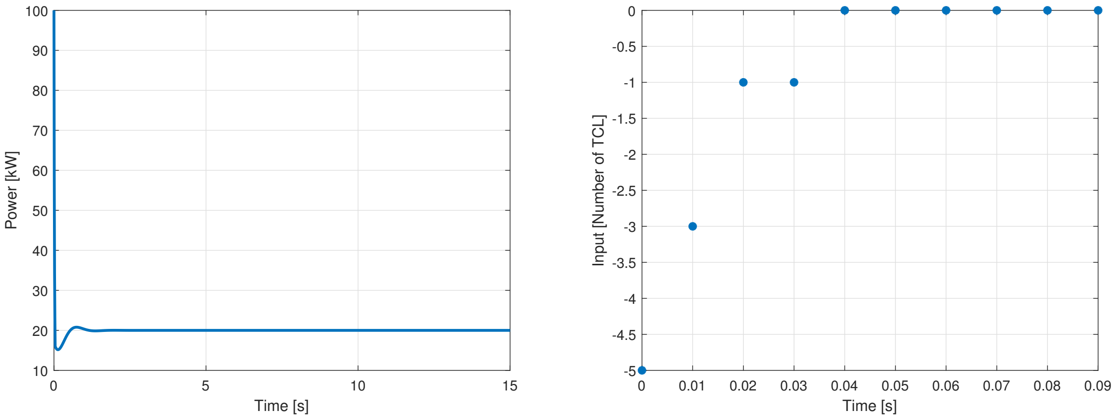


Figure 5-18: Output trajectory and control input for N_{bin} 2

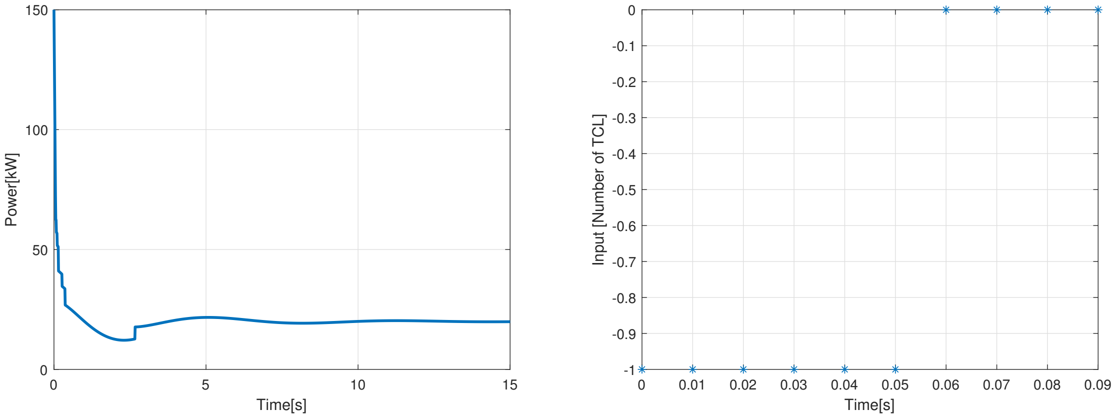


Figure 5-19: Output trajectory and control input for N_{bin} 3

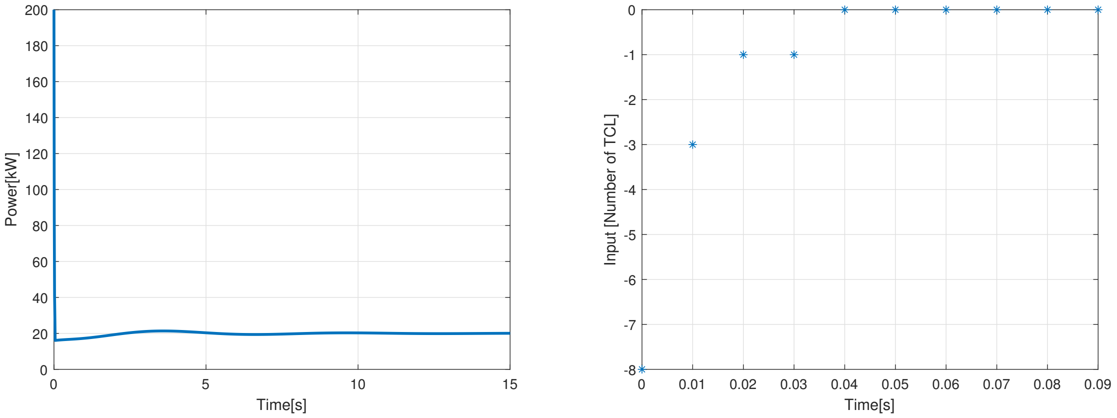


Figure 5-20: Output trajectory and control input for N_{bin} 4

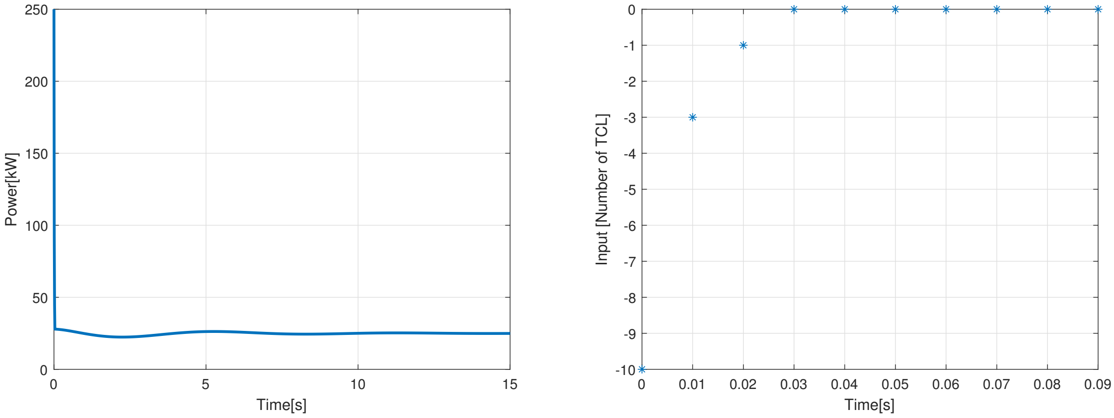


Figure 5-21: Output trajectory and control input for N_{bin} 5

Number of state bins N_{bin}	2	3	4	5
Time[ms]	8.50	33.78	43.28	80.07

Table 5-4: Computational complexity for varying state bins

In this chapter, a study is done to explore how the number of state bins and the heterogeneity affect the performance of an ensemble of TCL. The study gives a more acute understanding of the limitations of the model/algorithm and how it affects the performance.

Chapter 6

Conclusion and Future work

In this research thesis, an attempt is made to answer the following questions:

1. *Can adaptive optimal control be applied to Demand Side Management (DSM) of Thermostatically Controlled Load (TCL)? Which TCL models should be used?*

   Yes, adaptive optimal control can be applied to DSM of TCL. But the type of control algorithm to be used depends on the model used. Three state-of-the-art models for TCLs developed in [4], [18] and [5] are discussed briefly.

2. *What are its limitations? How can these limitations be overcome?*

   The model developed in [4] suffers from the limitations that the states are immeasurable (Chapter 2). Hence an Output Feedback (OPFB) algorithm can be used (Chapter 3).

   The model developed in [18] and [5] suffers from the limitation that the system knowledge must be known approximately. But this limitation can be overcome by using identification techniques (Chapter 4).

3. *How does the parameter heterogeneity affect the performance of the control?*

   A detailed study is done analyzing the performance for varying heterogeneity distribution, input weight, set-point and number of state bins (Chapter 5). From this study, it can be concluded that a model based on state bins (with a high number of state-bins) is a more accurate system representation and the non-linear control algorithm applied to it produces results with good performance.

There are a few interesting future works which can be investigated

- The system developed in section 4-2 is controlled using the non-linear control approach. The constraints that the system states face of being non-negative and the summation limits are now controlled only by increasing the weight on the input (R). But an interesting alternative approach would be to look into the direction of adaptive optimal control where the constraints are also included in the learning process [41].

- All the control procedures carried out in this research thesis is in Continuous Time (CT). Since all the systems, in reality, are more discrete in nature, a more appropriate control methodology would be to look into the application of Discrete Time (DT) control policies to the system developed in section 4-2. A possible direction to look into would be along the methodologies stated in [42].

- The system studied in this research thesis is considered with no external disturbances. But there can be possible external disturbances like the ambient temperature, second-order dynamics along the TCL, etc. Hence a non-linear control approach with disturbances can be studied. [19] has some interesting possible approaches to this work.

- The reference signals considered in this study is a step change. An approach to look into is where the set-point changes in a sinusoidal or ramp fashion. It is important to study these types of signals as they have been studied in [4] and references therein

- The algorithm proposed in this thesis is a Policy Iteration (PI) adaptive optimal algorithm. To make it completely online, a Recursive Least Squares (RLS) can be performed as in [37]. A comparative study of time taken to converge and studying its performance would be interesting future work. It is interesting to look into this option as policy iteration makes the system converge faster.

Appendix A

Subspace Identification - Autonomous Systems

Consider a Linear Time Invariant (LTI) Discrete Time (DT) system

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{aligned}$$

where $x(k) \in \mathbb{R}^n, u(k) \in \mathbb{R}^m$ and $y(k) \in \mathbb{R}^l$. When the states and the outputs are measurable, the following data equation can be constructed as shown below. To use this relation in subspace identification it is necessary that $s > n$.

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ \vdots \\ y(s-1) \end{bmatrix} = \underbrace{\begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{s-1} \end{bmatrix}}_{O_s} x(0) + \underbrace{\begin{bmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ CA^{s-2}B & CA^{s-3}B & \dots & CB & D \end{bmatrix}}_{T_s} \begin{bmatrix} u(0) \\ u(1) \\ u(2) \\ \vdots \\ u(s-1) \end{bmatrix}$$

To make use of more data points, a *Hankel* matrix can be constructed as shown below. And in general, we have $n < s << N$.

$$\underbrace{\begin{bmatrix} y(0) & y(1) & \dots & y(N-1) \\ y(1) & y(2) & \dots & y(N) \\ y(2) & y(3) & \dots & y(N+1) \\ \vdots & \vdots & \dots & \vdots \\ y(s-1) & y(s) & \dots & y(N+s-2) \end{bmatrix}}_{Y_{0.s.N}} = O_s X_{0,N} + T_s \underbrace{\begin{bmatrix} u(0) & u(1) & \dots & u(N-1) \\ u(1) & u(2) & \dots & u(N) \\ u(2) & u(3) & \dots & u(N+1) \\ \vdots & \vdots & \dots & \vdots \\ u(s-1) & u(s) & \dots & u(N+s-2) \end{bmatrix}}_{U_{0.s.N}}$$

where

$$X_{0,N} = \begin{bmatrix} x(0) & x(1) & \dots & x(N-1) \end{bmatrix}$$

Hence the data equation becomes as represented below. The explanation of the subscripts is: the first entry represents the time index of the first element of the matrix, the second and the third entry represents the number of rows and columns of the matrix.

$$Y_{0,s,N} = O_s X_{0,N} + T_s U_{0,s,N}$$

For an autonomous system, all the entries of $U_{0.s.N}$ is zero. Hence the data equation becomes

$$Y_{0,s,N} = O_s X_{0,N}$$

Let the Singular Value Decomposition (SVD) of $Y_{0,s,N}$ be represented by

$$Y_{0,s,N} = U_n \Sigma_n V_n^T,$$

where $\Sigma_n \in \mathbb{R}^{n \times n}$ and $rank(\Sigma_n) - n$, then U_n can be denoted by

$$U_n = O_s T = \begin{bmatrix} C_T \\ C_T A_T \\ \vdots \\ C_T A_T^{s-1} \end{bmatrix}.$$

The matrix A_T is computed by solving the set of equations and it has a unique solution.

$$U_n(1:(s-1)l,:)A_T = U_n(l+1:sl,:)$$

Here A_T represents the matrix calculated upto similarity transformation. The matrix A and A_T have the same eigenvalues which determine the system dynamics but have different matrix elements.

Bibliography

[1] "GWEC global wind statistics." https://gwec.net/wp-content/uploads/vip/GWEC_PRstats2017_EN-003_FINAL.pdf. Accessed: 2019-05-05.

[2] "International energy agency." https://www.iea.org/newsroom/energysnapshots/air-conditioner-sales-growth.html. Accessed: 2019-05-05.

[3] J. L. Mathieu, P. N. Price, S. Kiliccote, and M. A. Piette, "Quantifying changes in building electricity use, with application to demand response," *IEEE Transactions on Smart Grid*, vol. 2, no. 3, pp. 507–518, 2011.

[4] S. Kundu, N. Sinitsyn, S. Backhaus, and I. Hiskens, "Modeling and control of thermostatically controlled loads," *arXiv preprint arXiv:1101.2157*, 2011.

[5] S. Bashash and H. K. Fathy, "Modeling and control insights into demand-side energy management through setpoint control of thermostatic loads," in *American Control Conference (ACC), 2011*, pp. 4546–4553, IEEE, 2011.

[6] Y. Jiang and Z.-P. Jiang, *Robust Adaptive Dynamic Programming*. John Wiley & Sons, 2017.

[7] "International energy agency." https://www.iea.org/renewables2018/. Accessed: 2019-05-05.

[8] Y. V. Makarov, C. Loutan, J. Ma, and P. De Mello, "Operational impacts of wind generation on california power systems," *IEEE Transactions on Power Systems*, vol. 24, no. 2, pp. 1039–1050, 2009.

[9] N. Lu, "An evaluation of the HVAC load potential for providing load balancing service," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1263–1270, 2012.

[10] D. S. Callaway, "Tapping the energy storage potential in electric loads to deliver load following and regulation, with application to wind energy," *Energy Conversion and Management*, vol. 50, no. 5, pp. 1389–1400, 2009.

[11] S. Koch, M. Zima, and G. Andersson, "Potentials and applications of coordinated groups of thermal household appliances for power system control purposes," in *2009 IEEE PES/IAS Conference on Sustainable Alternative Energy (SAE)*, pp. 1–8, IEEE, 2009.

[12] N. Lu and S. Katipamula, "Control strategies of thermostatically controlled appliances in a competitive electricity market," in *IEEE Power Engineering Society General Meeting, 2005*, pp. 202–207, IEEE, 2005.

[13] M. Gustafson, J. Baylor, and G. Epstein, "Direct water heater load control-estimating program effectiveness using an engineering model," *IEEE Transactions on Power Systems*, vol. 8, no. 1, pp. 137–143, 1993.

[14] T. Ericson, "Direct load control of residential water heaters," *Energy Policy*, vol. 37, no. 9, pp. 3502–3512, 2009.

[15] S. Baldi, A. Karagevrekis, I. T. Michailidis, and E. B. Kosmatopoulos, "Joint energy demand and thermal comfort optimization in photovoltaic-equipped interconnected microgrids," *Energy Conversion and Management*, vol. 101, pp. 352 – 363, 2015.

[16] C. D. Korkas, S. Baldi, and E. B. Kosmatopoulos, "9 - grid-connected microgrids: Demand management via distributed control and human-in-the-loop optimization," in *Advances in Renewable Energies and Power Technologies* (I. Yahyaoui, ed.), pp. 315 – 344, Elsevier, 2018.

[17] J. Taneja, D. Culler, and P. Dutta, "Towards cooperative grids: Sensor/actuator networks for renewables integration," in *2010 First IEEE International Conference on Smart Grid Communications*, pp. 531–536, IEEE, 2010.

[18] S. Koch, J. L. Mathieu, and D. S. Callaway, "Modeling and control of aggregated heterogeneous thermostatically controlled loads for ancillary services," in *Proc. PSCC*, pp. 1–7, Citeseer, 2011.

[19] M. Liu and Y. Shi, "Model predictive control of aggregated heterogeneous second-order thermostatically controlled loads for ancillary services," *IEEE Transactions on Power Systems*, vol. 31, no. 3, pp. 1963–1971, 2016.

[20] M. Chertkov and V. Chernyak, "Ensemble of thermostatically controlled loads: statistical physics approach," *Scientific Reports*, vol. 7, no. 1, p. 8673, 2017.

[21] A. Ghaffari, S. Moura, and M. Krstić, "Modeling, control, and stability analysis of heterogeneous thermostatically controlled load populations using partial differential equations," *Journal of Dynamic Systems, Measurement, and Control*, vol. 137, no. 10, p. 101009, 2015.

[22] B. J. Claessens, D. Vanhoudt, J. Desmedt, and F. Ruelens, "Model-free control of thermostatically controlled loads connected to a district heating network," *Energy and Buildings*, vol. 159, pp. 1–10, 2018.

[23] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits and Systems Magazine*, vol. 9, no. 3, 2009.

[24] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis, "Adaptive optimal control for continuous-time linear systems based on policy iteration," *Automatica*, vol. 45, no. 2, pp. 477–484, 2009.

[25] H. Modares and F. L. Lewis, "Linear quadratic tracking control of partially-unknown continuous-time systems using reinforcement learning," *IEEE Transactions on Automatic control*, vol. 59, no. 11, pp. 3051–3056, 2014.

[26] B. Luo and H.-N. Wu, "Online adaptive optimal control for bilinear systems," in *2012 American Control Conference (ACC)*, pp. 5507–5512, IEEE, 2012.

[27] Y. Jiang and Z. P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, 2012.

[28] T. Bian and Z. P. Jiang, "Value iteration and adaptive dynamic programming for data-driven adaptive optimal control design," *Automatica*, vol. 71, pp. 348–360, 2016.

[29] T. Bian and Z. P. Jiang, "Stochastic and adaptive optimal control of uncertain interconnected systems: A data-driven approach," *Systems and Control Letters*, vol. 115, pp. 48–54, 2018.

[30] W. Gao and Z.-P. Jiang, "Adaptive dynamic programming and adaptive optimal output regulation of linear systems," *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 4164–4169, 2016.

[31] H. Modares, F. L. Lewis, and Z. P. Jiang, "Optimal Output-Feedback Control of Unknown Continuous-Time Linear Systems Using Off-policy Reinforcement Learning," *IEEE Transactions on Cybernetics*, vol. 46, no. 11, pp. 2401–2410, 2016.

[32] Y. Jiang and Z.-P. Jiang, "Robust Adaptive Dynamic Programming for Large-Scale Systems With an Application to Multimachine Power Systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 59, no. 10, pp. 693–697, 2012.

[33] Y. Jiang and Z.-P. Jiang, "Robust adaptive dynamic programming with an application to power systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 7, pp. 1150–1156, 2013.

[34] Y. Jiang and Z.-P. Jiang, "Robust approximate dynamic programming and global stabilization with nonlinear dynamic uncertainties," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 115–120, IEEE, 2011.

[35] L. Chang, X. Wang, and M. Mao, "Forecast of schedulable capacity for thermostatically controlled loads with big data analysis," in *Power Electronics for Distributed Generation Systems (PEDG), 2017 IEEE 8th International Symposium on*, pp. 1–6, IEEE, 2017.

[36] D. Kleinman, "On an iterative technique for Riccati equation computations," *IEEE Transactions on Automatic Control*, vol. 13, no. 1, pp. 114–115, 1968.

[37] P. A. Ioannou and J. Sun, *Robust adaptive control.* Courier Corporation, 2012.

[38] M. Verhaegen and V. Verdult, *Filtering and system identification: a least squares approach.* Cambridge university press, 2007.

[39] Z.-P. Jiang, A. R. Teel, and L. Praly, "Small-gain theorem for iss systems and applications," *Mathematics of Control, Signals and Systems*, vol. 7, no. 2, pp. 95–120, 1994.

[40] M. J. D. Powell, *Approximation theory and methods.* Cambridge university press, 1981.

[41] H. Modares, F. L. Lewis, and M.-B. Naghibi-Sistani, "Adaptive optimal control of unknown constrained-input systems using policy iteration and neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 10, pp. 1513–1525, 2013.

[42] D. Wang, D. Liu, Q. Wei, D. Zhao, and N. Jin, "Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming," *Automatica*, vol. 48, no. 8, pp. 1825–1832, 2012.

Glossary

List of Acronyms

ADP	Adaptive Dynamic Programming
ARE	Algebraic Riccati Equation
CT	Continuous Time
DSM	Demand Side Management
DT	Discrete Time
GAS	Globally Asymptotically Stable
IRL	Integral Reinforcement Learning
ISS	Input to State Stable
LQI	Linear Quadratic Integral
LQR	Linear Quadratic Regulator
LQT	Linear Quadratic Tracking
LS	Least Squares
LTI	Linear Time Invariant
MPC	Model Predictive Control
OPFB	Output Feedback
PI	Policy Iteration
RLS	Recursive Least Squares
RQ	Research Question
SFB	State Feedback

SISO	Single Input Single Output
SVD	Singular Value Decomposition
TCL	Thermostatically Controlled Load
VI	Value Iteration