

# Building Temperature Distributed Control via Explicit MPC and “Trim and Respond” Methods

Sarah Koehler\*, Francesco Borrelli\*

**Abstract**—We study the distributed control of heating, ventilation, and air conditioning systems in buildings. We present and compare two control design techniques. In the first part, a one-step distributed model predictive controller (DMPC) is introduced. We compute the explicit state-feedback solution and show how to implement it on existing networked buildings control platforms. In the second part, we discuss a control logic currently used in the buildings industry called “Trim and Respond”. We show that “Trim and Respond” controllers can be seen as a special class of one-step DMPC algorithms. We conclude the paper with a simulation study which shows the DMPC control design and its equivalence with “Trim and Respond” heuristics.

## I. INTRODUCTION

The building sector consumes about 40% of the energy used in the United States and is responsible for nearly 40% of greenhouse gas emissions [1]. It is therefore economically, socially, and environmentally significant to reduce the energy consumption of buildings.

The main idea of predictive control is to use a model of the plant to predict the future evolution of the system [2], [3], [4]. At each sampling time, starting at the current state, an open-loop optimal control problem is solved over a finite horizon. The optimal command signal is applied to the process only during the following sampling interval. At the next time step, a new optimal control problem based on new measurements of the state is solved over a shifted horizon. The resultant control algorithm is referred to as Model Predictive Control (MPC).

In [5], [6], [7], [8], the authors have implemented MPC on building heating, ventilation, and air conditioning (HVAC) systems obtaining reduced peak power consumption and reduced overall energy usage compared to existing production logic. In the aforementioned literature, control computation is performed at a centralized unit. In practice, embedded building control platforms are physically distributed throughout the building. It is standard to have a central processor and router located at the main air supply location (called the Air Handling Unit) and an embedded control unit at the locations which regulate flow and temperature of each zone (called the Variable Air Volume box). The goal of this paper is to design distributed MPC control algorithms which fit these existing distributed control platforms.

Distributed Model Predictive Control (DMPC) in buildings with a large number of zones has been previously approached by using Sequential Quadratic Programming (SQP) and dual

decomposition [9]. The control scheme is formulated for a nonlinear system using MPC and a generic prediction horizon  $N$ . The result is an iterative decentralized controller with substantial communication overhead.

In this paper, we present and compare two distributed control design approaches: a one-step distributed MPC and a heuristic control logic called “Trim and Respond”. “Trim and Respond” is a distributed control logic currently utilized in buildings [10]; it has been shown to be more efficient and is considered to be more intuitive to tune than control logic used by equipment manufacturers.

The paper is divided into two parts. In the first part, we compute the explicit solution of a one-step distributed MPC controller and show how to implement it on existing networked buildings control platforms. In the second part, we present the “Trim and Respond” logic and compare it with the explicit solution of a one-step DMPC. We show that “Trim and Respond” is a special class of one-step DMPC. The results of this paper can impact the way distributed building control logics are designed. On one hand, they allow the implementation of DMPC on existing distributed building control platforms with a small amount of communication; on the other hand, they can be used to improve “Trim and Respond” logic by decreasing the amount of manual parameter tuning.

The paper is organized in the following way. First, we introduce the system model in Section II. Then, the constrained finite time optimization problem for MPC is formulated in Section III. The distributed explicit solution is derived from KKT conditions in Section IV. “Trim and Respond” logic is introduced in Section V. The DMPC and “Trim and Respond” methods are compared in Section VI, a simulation of both logics is shown in Section VII, and finally, conclusions are drawn in Section VIII.

## II. SYSTEM MODEL

This section introduces the components of a typical heating, ventilation, and air conditioning system. Air Handling Units (AHU) and Variable Air Volume (VAV) boxes are the main components used to heat or cool and distribute air in a building (Figure 1). The AHU recirculates return air from building spaces, and mixes it with fresh outside air. The proportion of return air to outside air is controlled by damper positions in the AHU (Figure 1). The mixed air can be cooled by cooling coils that extract the cooling energy from chilled water produced by chillers.

The air temperature after these coils depends on the coil valve positions, the temperature of the chilled or heated

\* S. Koehler and F. Borrelli are with the Department of Mechanical Engineering, University of California, Berkeley, CA 94720-1740, USA. E-mail: {skoehler, fborrelli}@berkeley.edu.

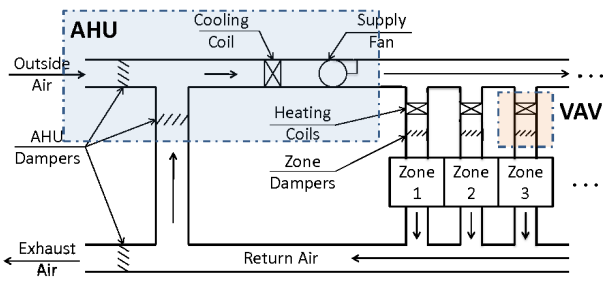


Fig. 1: System Scheme

water, the temperature of mixed air entering the cooling coil, the mass flow rate of the mixed air, and the physical characteristics as well as thermal effectiveness of the coils. Air is delivered to the building spaces by electrical fans. Before reaching a given space, the air goes through VAV boxes. At each VAV box, air temperature can be increased using reheat coils installed in the VAV box when needed. The space served by one VAV box is denoted by a thermal zone. The delivered air enters a zone through diffusers that are designed to fully mix the incoming air with the air in the thermal zone.

In this paper, the cooling power input provided by the AHU is denoted as  $u_c$ . The local reheating power input from the  $i$ -th VAV box is denoted by  $u_i$ . We assume that there are  $n$  thermal zones in the building. The system dynamics of zone  $i \in \{1, \dots, n\}$  are modeled using the following differential equation:

$$C_i \dot{T}_i = u_c + u_i + \frac{(T_{oa} - T_i)}{R_i} + P_{d,i}, \quad (1)$$

where  $T_i$  is the temperature of zone  $i$ ,  $T_{oa}$  is the outside air temperature,  $P_{d,i}$  is the load on zone  $i$  due to occupancy, weather, and other disturbances (e.g. thermal load from neighboring zones), and  $C_i$  and  $R_i$  are thermal resistive-capacitive parameters of the  $i$ -th zone. We remark that the system dynamics are generally nonlinear [7]. The model (1) is assumed to be a linearized model around nominal operating conditions.

Our notation highlights that  $u_c$  is computed and implemented at the central AHU level, whereas the local reheating control  $u_i$  is computed and implemented at the local VAV box. Note that an embedded processor exists at each VAV box and at the AHU.

The system dynamics of equation (1) are discretized using the Euler forward discretization method to obtain

$$T_i(k+1) = A_i T_i(k) + B_{C,i} u_c(k) + B_i u_i(k) + d_i(k), \quad (2)$$

where  $k$  is the time step,  $A_i = 1 - \frac{\Delta t}{R_i C_i}$ ,  $B_{C,i} = \frac{\Delta t}{C_i}$ ,  $B_i = \frac{\Delta t}{C_i}$ , and  $d_i = \frac{P_{d,i} \Delta t}{C_i} + \frac{T_{oa} \Delta t}{R_i C_i}$ . A study on the effect of discretization methods in MPC can be found in [11].

We model comfort constraints as thermal bounds on the temperature at the next time step:

$$\underline{T} \leq T_i(k+1) \leq \bar{T}. \quad (3)$$

Also, local actuation can only be in the form of heating and central actuation can only be in the form of cooling:

$$u_i(k) \geq 0, u_c(k) \leq 0. \quad (4)$$

Input saturation of  $u_c(k)$  and  $u_i(k)$  is not considered in this paper. The approach presented here can be extended to include these additional constraints.

### III. MODEL PREDICTIVE CONTROL

In this section, a model predictive controller is designed to minimize the total energy consumption of the HVAC system defined in Section II, and to satisfy the thermal comfort constraints in equation (3) and the reheating and cooling constraints (4). Consider the following MPC problem at time step  $k$ :

$$\min_{u_c(k), u_i(k)} u_c(k)^2 + \sum_{i=1}^n u_i(k)^2 \quad (5a)$$

subject to

$$A_i T_i(k) + B_{C,i} u_c(k) + B_i u_i(k) + \hat{d}_i(k) \leq \bar{T} \quad (5b)$$

$$A_i T_i(k) + B_{C,i} u_c(k) + B_i u_i(k) + \hat{d}_i(k) \geq \underline{T} \quad (5c)$$

$$u_i(k) \geq 0 \quad (5d)$$

$$u_c(k) \leq 0 \quad (5e)$$

$$\forall i \in \{1, \dots, n\},$$

where the prediction horizon length is one,  $\hat{d}_i(k)$  is the estimated load at the  $i$ -th zone, (5b) and (5c) are thermal comfort constraints at the next time step  $k+1$ , (5d) is the local actuation reheating constraint, and (5e) is the central cooling constraint.

Let the optimal solution to problem (5) be

$$\mathbf{U}^* = \{u_c^*(k), u_1^*(k), \dots, u_n^*(k)\}. \quad (6)$$

At time step  $k$ , the solution  $u_c^*(k)$  is implemented by the AHU and  $u_i^*(k)$  is implemented by the  $i$ -th VAV box. The optimization (5) is repeated at time  $k+1$ , with the updated state estimation  $T_i(k+1)$  and estimated load  $\hat{d}_i(k+1)$ , yielding a receding horizon control strategy.

### IV. EXPLICIT DISTRIBUTED MPC SOLUTION

The solution  $\mathbf{U}^*$  in (6) can be computed by solving a quadratic program at each time step  $k$  for a given  $T_1(k), \hat{d}_1(k), \dots, T_n(k), \hat{d}_n(k)$ . However, this would result in a centralized approach and would require software not readily available on current buildings embedded control systems.

Karush-Kuhn-Tucker (KKT) necessary and sufficient conditions are used to find the explicit expression of the optimal controller solution to problem (5). Compared to standard explicit MPC control design [12], our procedure provides a distributed explicit controller. The Lagrangian for problem (5) is

$$\begin{aligned} L(u_c(k), u_1(k), \dots, u_n(k), \lambda_c(k), \lambda_1(k), \dots, \lambda_n(k), \\ \bar{\mu}_1(k), \dots, \bar{\mu}_n(k), \underline{\mu}_1(k), \dots, \underline{\mu}_n(k)) = \\ u_c(k)^2 + \lambda_c(k)(u_c(k)) + \sum_{i=1}^n \left( u_i(k)^2 + \lambda_i(k)(-u_i(k)) \right. \\ \left. + \bar{\mu}_i(k)(A_i T_i(k) + B_{C,i} u_c(k) + B_i u_i(k) + \hat{d}_i(k) - \bar{T}) \right. \\ \left. + \underline{\mu}_i(k)(-A_i T_i(k) - B_{C,i} u_c(k) - B_i u_i(k) - \hat{d}_i(k) + \underline{T}) \right). \end{aligned} \quad (7)$$

Three dual variables exist for each zone  $i$ :  $\lambda_i(k)$  is associated with the non-negativity constraint on  $u_i(k)$  (5d),  $\bar{\mu}_i(k)$  is associated with the upper bound on the temperature  $T_i(k+1)$  (5b), and  $\underline{\mu}_i(k)$  is associated with the lower bound on the temperature  $\bar{T}_i(k+1)$  (5c). Also,  $\lambda_c(k)$  corresponds to the non-positivity constraint on  $u_c(k)$  (5e). The KKT conditions for problem (5) at time  $k$  are:

$$2u_c(k) + \sum_{i=1}^n B_{C,i}(\bar{\mu}_i - \underline{\mu}_i) + \lambda_c(k) = 0 \quad (8a)$$

$$2u_i(k) - \lambda_i(k) + B_i(\bar{\mu}_i - \underline{\mu}_i) = 0 \quad (8b)$$

$$\lambda_c(k)u_c(k) = 0 \quad (8c)$$

$$\lambda_i(k)u_i(k) = 0 \quad (8d)$$

$$\bar{\mu}_i(k)(A_iT_i(k) + B_{C,i}u_c(k) + B_iu_i(k) + \hat{d}_i(k) - \bar{T}) = 0 \quad (8e)$$

$$\underline{\mu}_i(k)(A_iT_i(k) + B_{C,i}u_c(k) + B_iu_i(k) + \hat{d}_i(k) - \underline{T}) = 0 \quad (8f)$$

$$u_c(k) \leq 0 \quad (8g)$$

$$u_i(k) \geq 0 \quad (8h)$$

$$A_iT_i(k) + B_{C,i}u_c(k) + B_iu_i(k) + \hat{d}_i(k) \leq \bar{T} \quad (8i)$$

$$A_iT_i(k) + B_{C,i}u_c(k) + B_iu_i(k) + \hat{d}_i(k) \geq \underline{T} \quad (8j)$$

$$\lambda_c(k) \geq 0 \quad (8k)$$

$$\lambda_i(k) \geq 0 \quad (8l)$$

$$\underline{\mu}_i(k) \geq 0 \quad (8m)$$

$$\bar{\mu}_i(k) \geq 0 \quad (8n)$$

$$i \in \{1, \dots, n\},$$

where equations (8a) to (8b) are the components of the KKT condition  $\nabla L(u_c(k), u_1(k), \dots) = 0$ , equations (8c) to (8f) are the complementary slackness conditions, equations (8g) to (8j) are the primal feasibility conditions, and equations (8k) to (8n) are the dual feasibility conditions.

In order to compute the feedback control law solving (8), we need to enumerate all possible combinations of active primal constraints. Intuition from physics makes this task easier. Only combinations of temperature bound constraints (8i) and (8j) need be considered, as the activeness of (8g) and (8h) are fully determined from active sets of (8i) and (8j). Let  $\mathcal{I}_H$  and  $\mathcal{I}_C$  be two subsets of the set  $\mathcal{I} = \{1, \dots, n\}$ . One can prove that only the following cases will generate a primal and dual feasible controller.

Case I Lower bound temperature constraints (8j) are active for all  $i \in \mathcal{I}_H$ ; upper bound temperature constraints (8i) are not active for all  $i \in \mathcal{I}$

Case II Upper bound temperature constraints (8i) are active for all  $i \in \mathcal{I}_C$ ; lower bound temperature constraints (8j) are not active for all  $i \in \mathcal{I}$

Case III Upper bound temperature constraints (8i) are active for all  $i \in \mathcal{I}_C$ ; lower bound temperature constraints (8j) are active for all  $i \in \mathcal{I}_H$ ;

Case IV Both Temperature bound constraints (8i) and (8j) are not active for all  $i \in \mathcal{I}$ .

Note that in Case III a single zone cannot simultaneously have an active upper temperature constraint and an active lower constraint, i.e.  $\mathcal{I}_H \cap \mathcal{I}_C = \emptyset$  in Case III.

**Remark 1.** All primal variables, dual variables, and inputs are a function of time step  $k$ , but for readability this argument will be dropped from the notation for the remainder of this section.

The primal variables can be written as a function of the

dual variables:

$$u_c = \frac{1}{2} \left( -\lambda_c + \sum_{i \in \mathcal{I}} B_{C,i}(\underline{\mu}_i - \bar{\mu}_i) \right) \quad (9a)$$

$$u_i = \frac{1}{2} \left( B_i(\underline{\mu}_i - \bar{\mu}_i) + \lambda_i \right). \quad (9b)$$

Note that the optimal control law for Case IV is  $u_c = u_i = 0 \forall i \in \mathcal{I}$ . The derivation of the optimal distributed controller for Case I is described next and a summary of the other cases are provided in the following subsections.

**A. Case I: Active Lower Bound Temperature Constraints and No Active Upper Bound Temperature Constraint**

In Case I we have:

$$\bar{\mu}_i = 0 \quad \forall i \in \mathcal{I} \quad (10a)$$

$$\underline{\mu}_i = 0 \quad \forall i \in \mathcal{I} \setminus \mathcal{I}_H \quad (10b)$$

$$A_iT_i + B_{C,i}u_c + B_iu_i + \hat{d}_i = \underline{T} \quad \forall i \in \mathcal{I}_H. \quad (10c)$$

Simplifying (9a) by using complimentary slackness (8d) with dual feasibility (8k) and (8m), we obtain:

$$u_c = 0, \quad (11a)$$

$$\lambda_c = \sum_{i \in \mathcal{I}_H} B_{C,i}\underline{\mu}_i. \quad (11b)$$

Thus, there is no cooling involved in Case I. Simplifying equation (9b) using complimentary slackness condition (8c), it is evident that  $\lambda_i = 0 \forall i \in \mathcal{I}$ . The local control law can be written as:

$$u_i = \frac{B_i}{2} \frac{\underline{\mu}_i}{\underline{\mu}_i} = \frac{\underline{T} - A_iT_i - \hat{d}_i}{B_i} \quad \forall i \in \mathcal{I}_H. \quad (12a)$$

$$u_i = 0 \quad \forall i \notin \mathcal{I}_H. \quad (12b)$$

In order for  $u_i \geq 0$  and  $\underline{\mu}_i \geq 0 \forall i \in \mathcal{I}_H$ , it is necessary that  $\underline{T} - A_iT_i - \hat{d}_i \geq 0$  (note that  $B_i > 0$ ). Also, if  $\underline{T} - A_iT_i - \hat{d}_i = 0 \forall i \in \mathcal{I}_H$ , the lower temperature bound is weakly active  $\forall i \in \mathcal{I}_H$  and the control law corresponds to the Case IV control law.

**Remark 2.** Intuitively, the optimality region of the Case I control law can be thought of as a non-cooling region because  $u_i \geq 0$  and  $u_c = 0$ . The zones  $i \in \mathcal{I}_H$  must request heat because they would violate the lower temperature bound on  $T_i(k+1)$  if  $u_c = u_i = 0$ .

**B. Case II: Active Upper Bound Temperature Constraints and No Active Lower Bound Temperature Constraint**

In Case II we have:

$$\underline{\mu}_i = 0 \quad \forall i \in \mathcal{I} \quad (13a)$$

$$\bar{\mu}_i = 0 \quad \forall i \in \mathcal{I} \setminus \mathcal{I}_C \quad (13b)$$

$$A_iT_i + B_{C,i}u_c + B_iu_i + \hat{d}_i = \bar{T} \quad \forall i \in \mathcal{I}_C. \quad (13c)$$

Following a similar approach to Case I, we obtain:

$$u_c = -\frac{(A_iT_i + \hat{d}_i - \bar{T})}{B_{C,i}} \quad \forall i \in \mathcal{I}_C \quad (14a)$$

$$u_i = 0 \quad \forall i \in \mathcal{I}. \quad (14b)$$

Note that  $u_c$  must be the same  $\forall i \in \mathcal{I}_C$  and  $(A_iT_i + \hat{d}_i - \bar{T}) \geq 0$  if  $i \in \mathcal{I}_C$ . If  $u_c = 0$ , the constraints

here are weakly active and the optimal controller is the same as in Case IV.

**Remark 3.** *Intuitively, this region is a non-heating region because  $u_c \leq 0$  and  $u_i = 0 \forall i \in \mathcal{I}$ .*

*C. Case III: Active Upper Bound Temperature Constraints and Active Lower Bound Temperature Constraints*

In Case III we have:

$$\bar{\mu}_i = 0 \quad \forall i \in \mathcal{I} \setminus \mathcal{I}_C \quad (15a)$$

$$A_i T_i + B_{C,i} u_c + B_i u_i + \hat{d}_i = \bar{T} \quad \forall i \in \mathcal{I}_C \quad (15b)$$

$$\underline{\mu}_i = 0 \quad \forall i \in \mathcal{I} \setminus \mathcal{I}_H \quad (15c)$$

$$A_i T_i + B_{C,i} u_c + B_i u_i + \hat{d}_i = \underline{T} \quad \forall i \in \mathcal{I}_H. \quad (15d)$$

By following the same approach as Case I and Case II, we can compute the optimal control law for Case III.

$$u_c = -\frac{(A_i T_i + \hat{d}_i - \bar{T})}{B_{C,i}} \quad \forall i \in \mathcal{I}_C \quad (16a)$$

$$u_i = 0 \quad \forall i \in \mathcal{I} \setminus \mathcal{I}_H \quad (16b)$$

$$u_i = \frac{\underline{T} - A_i T_i - \hat{d}_i - B_{C,i} u_c}{B_i} \quad \forall i \in \mathcal{I}_H. \quad (16c)$$

**Remark 4.** *Intuitively, this region can be thought of as a mixed heating and cooling region because  $u_c \leq 0$  and  $u_i \geq 0 \forall i \in \mathcal{I}_H$ .*

*D. Summary of Distributed Feedback Control Algorithm*

For all cases, it is convenient to define an initial guess for the dual variables  $\underline{\mu}_i$  and  $\bar{\mu}_i$  as follows:

$$\underline{\mu}_i^0 = (\underline{T} - A_i T_i - \hat{d}_i)_+ \quad (17a)$$

$$\bar{\mu}_i^0 = (-\bar{T} + A_i T_i + \hat{d}_i)_+, \quad (17b)$$

where  $(x)_+ := \max(x, 0)$ . Zone  $i$  requests heat if  $\underline{\mu}_i^0 > 0$ , and likewise,  $\bar{\mu}_i^0 > 0$  indicates that zone  $i$  requests cooling. In fact, given parameter values  $T_i$  and  $\hat{d}_i$  for all  $i \in \mathcal{I}$ , the sets  $\mathcal{I}_C$  and  $\mathcal{I}_H$  can be determined as follows:

$$\mathcal{I}_C = \left\{ i \in \mathcal{I} \mid i = \arg \min_i \frac{-\bar{\mu}_i^0}{B_{C,i}} \right\} \quad (18)$$

where  $\mathcal{I}_C = \emptyset$  if the minimizing value is zero, and

$$\mathcal{I}_H = \left\{ i \in \mathcal{I} \mid \underline{\mu}_i^0 - B_{C,i} u_c > 0 \right\}. \quad (19)$$

The zones in  $\mathcal{I}_C$  correspond to the zones requesting the maximum cooling required for any zone to maintain the upper temperature bound constraint (8i). Similarly, the zones in  $\mathcal{I}_H$  request heat in order to maintain the lower temperature bound constraint (8j), assuming  $u_c$  has been calculated. Case I has  $\mathcal{I}_C = \emptyset$ , Case II has  $\mathcal{I}_H = \emptyset$ , and Case IV has  $\mathcal{I}_C = \mathcal{I}_H = \emptyset$ .

Because the optimal law for  $u_c$  is determined from a sorting of  $\bar{\mu}_i^0$  and does not rely on  $u_i$ , only one local-to-central and one central-to-local communication is required. Therefore, the structure of this problem allows us to design a dual decomposition algorithm which converges in one step. The one-step DMPC solution is provided in Algorithm 1. Throughout the algorithm, it is assumed that each zone only

knows its own model parameters  $(A_i, B_{C,i}, B_i, \hat{d}_i)$ , and the central processor knows  $B_{C,i}$ . The keyword ‘‘Local’’ indicates a computation at the VAV processor, and the keyword ‘‘Central’’ indicates a computation at the AHU processor.

---

**Algorithm 1** Explicit Solution of one-step DMPC

---

**Computation of initial dual guess:**

---

- 1: Local: Calculate  $\underline{\mu}_i^0$  from (17a) and  $\bar{\mu}_i^0$  from (17b)
- 2: Send  $\underline{\mu}_i^0$  and  $\bar{\mu}_i^0$  to central AHU processor.

**Computation of Optimal Controller** :

---

- 1: Central: Compute  $\mathcal{I}_C$  from (18)
  - 2: **if**  $\mathcal{I}_C == \emptyset$  **then**
  - 3:     Central:  $u_c := 0$
  - 4: **else**
  - 5:     Central: Compute  $u_c$  from (16a)
  - 6: **end if**
  - 7: Send  $u_c$  to local VAV processors.
  - 8: Local: Compute if  $i \in \mathcal{I}_H$  from (19)
  - 9: **if**  $i \in \mathcal{I}_H$  **then**
  - 10:     Local: Compute  $u_i$  from (16c)
  - 11: **else**
  - 12:     Local:  $u_i := 0$ .
  - 13: **end if**
- 

**Remark 5.** *The Explicit DMPC solution presented in this Section IV is a fixed point solution of the dual-decomposition algorithm [9]. In fact, at each time step  $k$  the iterative dual-decomposition algorithm converges to one of the optimal controllers (Cases I-IV) presented in Section IV.*

## V. TRIM AND RESPOND

A well-regarded example of distributed control logic used in buildings is the ‘‘Trim and Respond’’ logic [10]. In this logic, each reheating input  $u_i(k)$  is controlled locally by a PI loop regulating zone temperature to remain within comfort bounds. Each zone generates a heating request when the reheating coil is approaching its maximum capacity. This maximum capacity is denoted as  $\bar{u}(k)$  and is changed over time by the ‘‘Trim and Respond’’ logic. Likewise, each zone generates a cooling request when the zone is close to its lower thermal bound. Since there is no cooling capacity at each zone  $i$ , the zone must request that the central cooling input  $u_c(k)$  add more cool air to the system, and thus  $u_c(k)$  is also modulated over time by the ‘‘Trim and Respond’’ logic as a function of cooling requests.

Next, we provide more details about the ‘‘Trim and Respond’’ logic. A heating request  $r_H^i(k)$  is generated by zone  $i$  at time step  $k$  if its local reheat control  $u_i(k)$  is near its maximum capacity  $\bar{u}(k)$ . Similarly, zone  $i$  generates a cooling request  $r_C^i(k)$  if its local reheating  $u_i(k)$  is near zero and the temperature is close to its upper bound  $\bar{T}$ . Each zone has an importance multiplier  $M_i \in \mathbb{Z}_{>0}$  which can be manually increased for a zone to have more influence on central control. The following formulas describe request

generation for zone  $i$ :

$$\text{If } u_i(k) \leq 0.05, T_i(k) \geq \bar{T} - 0.1, \quad r_c^i(k) = M_i. \quad (20a)$$

$$\text{If } u_i(k) \geq 0.95\bar{u}(k), \quad r_h^i(k) = M_i. \quad (20b)$$

If no cooling request is generated (conditional for (20a) is not met),  $r_c^i(k) = 0$ , and if no heating request is generated (conditional for (20b) is not met),  $r_h^i(k) = 0$ .

Requests are sent to centralized platforms associated with the main air handling unit and a central building boiler. The value for  $u_c(k)$  is reset at the AHU by adjusting the cooling coil valve position, and  $\bar{u}(k)$  is reset at the boiler by updating the supply water temperature that is provided to local reheating coils. In particular, the ‘‘Trim and Respond’’ logic responds to heating requests by decreasing  $u_c(k)$  by an amount proportional to the total number of cooling requests and by increasing  $\bar{u}(k)$  by an amount proportional to the number of heating requests.

At every time step  $k$ , regardless of the number of requests, the values of  $u_c(k)$ ,  $\bar{u}(k)$  are also ‘‘trimmed’’, or  $u_c(k)$  and  $\bar{u}(k)$  are decremented toward 0. Note that  $u_c(k)$  can only take on non-positive values whereas  $\bar{u}(k)$  can only take on non-negative values. This trimming effect ensures reduction in energy usage by decreasing centralized control efforts when no requests are made.

Let the sum of requests be  $\text{req}_c(k)$  and  $\text{req}_h(k)$ :

$$\text{req}_c(k) = \sum_{i=1}^n r_c^i(k), \quad \text{req}_h(k) = \sum_{i=1}^n r_h^i(k). \quad (21)$$

The central controls then ‘‘trim’’ and ‘‘respond’’ to these summed requests as follows:

$$\begin{aligned} &\text{In cooling mode, } (\text{req}_c(k) > \text{req}_h(k)) \\ u_c(k+1) &= u_c(k) + \text{trim}_c + (\text{req}_c(k) - r_I)\text{res}_c \end{aligned} \quad (22a)$$

$$\bar{u}(k+1) = \bar{u}(k) + \text{trim}_b. \quad (22b)$$

$$\begin{aligned} &\text{In heating mode, } (\text{req}_h(k) > \text{req}_c(k)) \\ u_c(k+1) &= u_c(k) + \text{trim}_c \end{aligned} \quad (23a)$$

$$\bar{u}(k+1) = \bar{u}(k) + \text{trim}_b + (\text{req}_h(k) - r_I)\text{res}_b. \quad (23b)$$

where the tuning parameters are:  $r_I$ , a fixed number of ignored requests;  $\text{res}_c$ ,  $\text{res}_b$ , the proportional ‘‘response’’ per request for  $u_c$  in cooling mode and  $\bar{u}$  in heating mode, respectively; and  $\text{trim}_c$ ,  $\text{trim}_b$  ‘‘trim’’ amounts. If  $\text{req}_c(k) = \text{req}_h(k)$ , then the control law uses (22b) and (23a).

The values  $u_c(k)$ ,  $\bar{u}(k)$ ,  $\text{req}_c(k)$  and  $\text{req}_h(k)$  are also limited by the following upper and lower bounds:

$$\text{req}_c(k) \leq \frac{\max_c}{\text{res}_c} + r_I, \quad \text{req}_h(k) \leq \frac{\max_b}{\text{res}_b} + r_I, \quad (24a)$$

$$u_c \in [u_{c_{\min}}, u_{c_{\max}}], \quad \bar{u} \in [0, \bar{u}_{\max}]. \quad (24b)$$

An example set of parameters used in cooling mode on an actual building is  $\text{trim}_c = 5/9^\circ\text{C}$ ,  $\text{res}_c = -10/9^\circ\text{C}$ ,  $u_{c_{\max}} = 0^\circ\text{C}$ ,  $u_{c_{\min}} = -12^\circ\text{C}$ ,  $\max_c = -4^\circ\text{C}$ ,  $r_I = 0$ .

**Remark 6.** *The logic presented in this section is a simplified version of ‘‘Trim and Respond’’ logic used in buildings today. In particular, we have only considered heating and cooling*

*requests. In practice, the logic extends to pressure control as well.*

## VI. COMPARISON OF METHODS

The explicit MPC method and the ‘‘Trim and Respond’’ method lead to controllers with the same structure and similar data communication. A detailed explanation is provided next.

The initial dual variables and requests have the same meaning: initial dual variables are a real-valued measurement of temperature bound violation (17), whereas requests (20) are integer-valued indicators of zone heating or cooling demand. While dual variables are the one-step predicted heating or cooling demand, requests are manually tuned to demand an appropriate amount of central control adjustment. The ‘‘Trim and Respond’’ algorithm handles this coarse measurement of demand by ignoring some requests (using  $\text{req}_I$ ) and setting a maximum change in centralized control, e.g.  $\max_C$  to prevent overreaction. If the model is accurate, the dual variables provide more valuable information than the boolean requests values.

The ‘‘Trim and Respond’’ importance factor  $M_i$  attempts to manually find the zone that has the highest cooling demand. Typically, one zone tends to request the most cooling (e.g. a computer server room or other zone with regularly high levels of thermal load), and so it is given the highest importance factor. The MPC algorithm automatically finds this zone by calculating  $\mathcal{I}_C$ .

Even if requests are only a function of the current state  $T(k)$ , predictive capability is introduced in practice. Requests are generated when local controls are close to their limits instead of at saturation. Thus, requests are generated when temperature bounds are likely to be saturated in the near future.

The explicit MPC algorithm and the ‘‘Trim and Respond’’ logics communicate between processors in a similar fashion. ‘‘Trim and Respond’’ sends information from local processors to centralized processors. Communication from central to local processors is not present because each local PI controller will implicitly respond to changes in  $u_c$ . Explicit MPC requires one communication each way between local and central processors (see Algorithm 1).

Both methods address energy efficiency. MPC directly addresses this by minimizing an effective cost of energy use. ‘‘Trim and Respond’’ addresses energy consumption by using a ‘‘trim’’ factor to decrease the control setpoints.

Finally, ‘‘Trim and Respond’’ does not require load estimation or a system dynamics model. However, there are many parameters to manually tune in the ‘‘Trim and Respond’’ algorithm.

## VII. RESULTS: SIMULATIONS

Simulation results are presented in this section for the Model Predictive Control algorithm and the ‘‘Trim and Respond’’ algorithm. The simulation is run with  $n = 10$  zones. The MPC has a sampling time of 5 minutes; the ‘‘Trim and Respond’’ logic updates the centralized controls  $u_c(k)$ ,  $\bar{u}(k)$  every 5 minutes with zone-level PI loops running

at 1 minute intervals. Model parameters from equation (1) are:  $C_i = 9.2e3kJ/^\circ C$ ,  $R_i = 50^\circ C/kW$ ,  $\underline{T} = 21^\circ C$ , and  $\bar{T} = 26^\circ C$ .

The disturbance  $d_i$  is shown in Figure 2, estimated from data from the UC Berkeley Bancroft Library. Simulation results are provided in Figure 3. The MPC implementation uses a Luenberger observer to estimate the load, allowing a slight violation of temperature bounds. The “Trim and Respond” profile shows many small variations in temperature, while MPC provides a smoother result. The small variations are an artifact of the integer requests and the prefixed resolution by which central controls can change in the “Trim and Respond” logic. Similar comparisons exist between the control inputs of the two methods as well as between the MPC initial dual variables and the “Trim and Respond” requests. A calculation of total power used illustrates that the explicit MPC outperforms the “Trim and Respond” method.

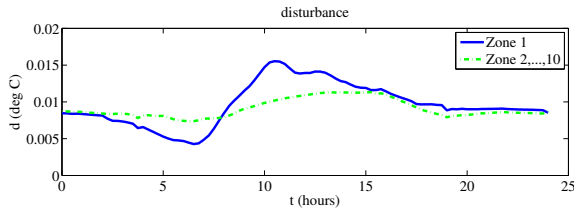


Fig. 2: Disturbance Profile  $d_i$  over one day

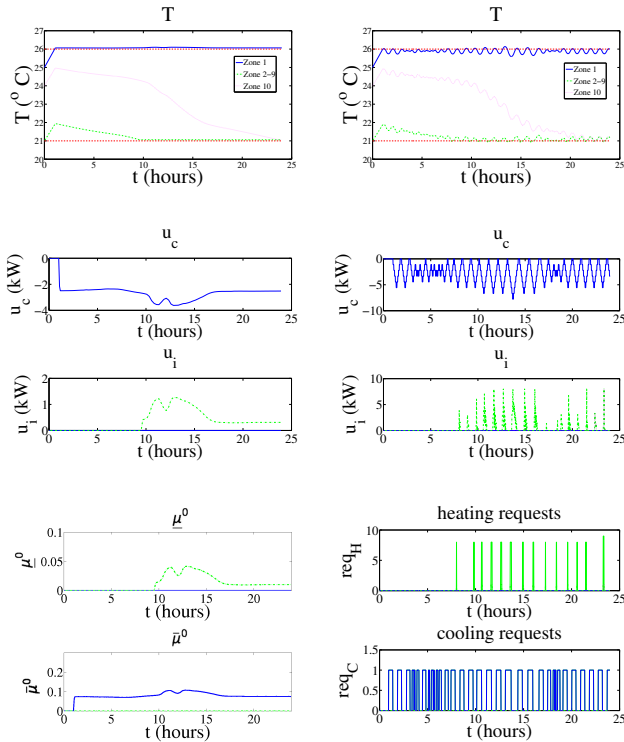


Fig. 3: Left column: MPC simulation results  
Right column: “Trim and Respond” simulation results

## VIII. CONCLUSION

In this paper, we have demonstrated two ways to approach the design of distributed control in building HVAC systems. A one-step DMPC was formulated and explicitly solved by using KKT conditions. It was then compared to an existing control strategy, “Trim and Respond”. Simulation results of both algorithms are very similar. Future work includes adding control saturation to the explicit solution and implementing the explicit MPC algorithm on hardware.

## ACKNOWLEDGEMENTS

We would like to thank Allan Daly for fruitful discussions regarding the “Trim and Respond” logic, and Yudong Ma for discussing dual decomposition in a building network. This material is based upon work partially supported by the National Science Foundation under Grant No. 0844456. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

## REFERENCES

- [1] J. McQuade, “A system approach to high performance buildings,” tech. rep., United Technologies Corporation, Feb. 2009. <http://gop.science.house.gov/Media/hearings/energy09/april28/mcquade.pdf>.
- [2] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, “Constrained model predictive control: Stability and optimality,” *Automatica*, vol. 36, no. 6, pp. 789 – 814, 2000.
- [3] F. Allgoewer, T. A. Badgwell, J. S. Qin, J. B. Rawlings, and S. J. Wright, “Nonlinear predictive control and moving horizon estimation—an introductory overview,” in *Advances in Control* (P. M. Frank, ed.), Springer, 1999.
- [4] J. H. Lee, M. Morari, and C. E. Garcia, “State-space interpretation of model predictive control,” *Automatica*, vol. 30, no. 4, pp. 707 – 717, 1994.
- [5] Y. Ma, F. Borrelli, B. Hency, B. Coffey, S. Bengea, and P. Haves, “Model predictive control for the operation of building cooling systems,” *Control Systems Technology, IEEE Transactions on*, vol. 20, pp. 796 – 803, may 2012.
- [6] F. Oldewurtel, A. Parisio, C. Jones, M. Morari, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, and K. Wirth, “Energy efficient building climate control using stochastic model predictive control and weather predictions,” in *American Control Conference (ACC), 2010*, pp. 5100 – 5105, 30 2010-july 2 2010.
- [7] Y. Ma, A. Kelman, A. Daly, and F. Borrelli, “Predictive control for energy efficient buildings with thermal storage: Modeling, stimulation, and experiments,” *Control Systems, IEEE*, vol. 32, pp. 44 – 64, feb. 2012.
- [8] S. C. Bengea, A. D. Kelman, F. Borrelli, R. Taylor, and S. Narayanan, “Model predictive control for mid-size commercial building hvac: Implementation, results and energy savings,” in *The Second International Conference on Building Energy and Environment*, 2012.
- [9] Y. Ma, G. Anderson, and F. Borrelli, “A distributed predictive control approach to building temperature regulation,” in *American Control Conference*, 2011.
- [10] S. Taylor, “Increasing efficiency with vav system static pressure setpoint reset,” *American Society of Heating, Refrigerating and Air-Conditioning Engineers*, june 2007.
- [11] A. Kelman, Y. Ma, and F. Borrelli, “Analysis of local optima in predictive control for energy efficient buildings,” *Journal of Building Performance Simulation*, Apr 2012.
- [12] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos, “The explicit solution of model predictive control via multiparametric quadratic programming,” in *American Control Conference, 2000. Proceedings of the 2000*, vol. 2, pp. 872 – 876 vol.2, 2000.