

Large Neighborhood Search for Energy Aware Meeting Scheduling in Smart Buildings

BoonPing Lim¹, Menkes van den Briel¹, Sylvie Thiébaux¹,
Russell Bent², and Scott Backhaus²

¹ NICTA and The Australian National University, Canberra, Australia

² Los Alamos National Laboratory, Los Alamos, United States

Abstract. One of the main inefficiencies in building management systems is the widespread use of schedule-based control when operating heating, ventilation and air conditioning (HVAC) systems. HVAC systems typically operate on a pre-designed schedule that heats or cools rooms in the building to a set temperature even when rooms are not being used. Occupants, however, influence the thermal behavior of buildings. As a result, using occupancy information for scheduling meetings to occur at specific times and in specific rooms has significant energy savings potential. As shown in Lim et al. [15], combining HVAC control with meeting scheduling can lead to substantial improvements in energy efficiency. We extend this work and develop an approach that scales to larger problems by combining mixed integer programming (MIP) with large neighborhood search (LNS). LNS is used to destroy part of the schedule and MIP is used to repair the schedule so as to minimize energy consumption. This approach is far more effective than solving the complete problem as a MIP problem. Our results show that solutions from the LNS-based approach are up to 36% better than the MIP-based approach when both given 15 minutes.

Keywords: smart buildings, scheduling, large neighborhood search, HVAC control

1 Introduction

In 2010, electricity expenditures in both residential and commercial buildings were over 300 billion dollars in the United States alone. About one third of this consumption can be attributed to space heating, ventilation and air conditioning (HVAC)³. Hence, with over 100 billion dollars in annual electricity expenditures even a small percentage improvement in the operation of HVAC systems can lead to significant savings.

While it is possible to reduce energy consumption by retrofitting buildings with more efficient HVAC systems, it is far more cost effective to improve their control algorithms. A building management system (BMS) predominantly operates on a pre-designed schedule that incorporates a nighttime setback strategy,

³ See Table 1.2.5 at <http://buildingsdatabook.eren.doe.gov/>

which relaxes the comfort constraints during the night. Bloomfield and Fisk [2] show that such a strategy leads to significant energy savings. Building management systems, however, could be far more dynamic by adopting strategies that consider occupancy information. Occupants impact the thermal behavior of buildings. For example, a BMS may relax the comfort constraints for those rooms in the building that are not planned on being used.

HVAC control and meeting scheduling have been studied extensively as separate problems and only recent work has started to look at integrating the two problems [15]. This work introduced a mixed integer programming (MIP) model for energy aware meeting scheduling. Given the highly-constrained nature of meeting scheduling with HVAC control, solving the integrated problem as a MIP seems to be a reasonable choice. MIP easily manages the interaction between meeting scheduling and the impact it has on HVAC energy consumption. The problem with the MIP-based approach, however, is that it does not scale. Consequently, in this work we introduce a hybrid solution that combines MIP with large neighborhood search (LNS) so as to scale to problem sizes that, for example, companies and universities may face when scheduling meetings and courses. In order to fine-tune the LNS heuristic we apply automatic parameter tuning, which is a useful step in producing better results on average.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 describes the MIP formulation, which consists of an HVAC model and a scheduling model that are combined into one energy aware meeting scheduling model. Section 4 describes the LNS heuristic and discusses the automatic parameter tuning. Experimental results are presented in Section 5 and we conclude with key observations and opportunities for further work in Section 6.

2 Related Work

Energy aware meeting scheduling has gotten more attention in recent years possibly due to the significant cost saving opportunities. Even simple rules like consolidating meetings in fewer buildings have proven to be very effective. For example, Portland State University consolidated night and weekend classes that were held across 21 buildings into 5 energy efficient buildings. By doing so, they reported a reduction of 18.5% in electricity consumption over the fall season compared to the previous three-year average [22].

Majumdar et al. [16] evaluate the performance of a number of scheduling algorithms that they run through the energy simulation software EnergyPlus [3]. Some criteria considered in their algorithms include minimizing the number of rooms, minimizing time between meetings in the same room, and minimizing room size. The latter performed best when considering only one criterion, but a more involved A* search with successive calls to EnergyPlus performed best overall. Similar criteria are considered in the work by Kwak and others [12, 13], but they use MIP to calculate an energy efficient schedule. Pan et al. [21] use the criterion of scheduling meetings back-to-back and observed 30% energy savings when comparing results to existing schedules.

It is important to note that these works minimize energy use without directly modeling the HVAC system. In our work we integrate meeting scheduling with HVAC control, because it is of key importance to determine which rooms need to be heated or cooled and when. For example, sometimes it is more energy efficient to pre-cool a room rather than cool it at the time of a meeting. In order to determine the HVAC operation one needs to decide the HVAC control settings.

Occupancy information, knowing whether or not there are people in a room and ideally also knowing how many people there are in a room, is at the basis of moving away from schedule-based control towards model predictive control (MPC). Schedule-based control is inefficient as it operates HVAC systems according to a fixed schedule that often assumes maximum zone occupancy. Oldewurtel et al. [20] use MPC, which takes into account weather and occupancy forecasts to estimate energy saving opportunities. Goyal et al. [10] also take into account occupancy forecasts and compare the performance of MPC strategies with feedback controllers using occupancy measurements. Their conclusion is that occupancy predictions can lead to additional energy savings over occupancy measurements especially when ventilation standards change. Currently the ASHRAE ventilation standard[1] requires that a certain amount of outside air is mixed with return air from the zones regardless whether a zone is occupied or not.

Lim et al. [15] combine MPC with meeting scheduling and propose a MIP-based solution. This model achieves an energy reduction of over 50% when compared to approaches similar to those presented in [10, 13, 16]. The drawback of this approach is that it does not scale well, which is why we developed a hybrid solution that combines MIP with LNS. In [15], we presented some of the initial results on applying LNS on this problem. Here, we significantly expand on the details of the LNS algorithm, and present extensive results supporting the quality of the algorithm.

Combining constraint-based methods with neighborhood search methods is not new. For example, LeBras et al. [14] used LNS with MIP in network design for a species conservation problem, Di Gaspero et al. [5] used LNS with CP in balancing bike sharing systems, Rendl et al. [23] applied CP with variable neighborhood search for homecare scheduling, and Danna et al. [4] used LNS in job-shop scheduling. We are, however, unaware of its application in the space of energy aware scheduling in smart buildings.

3 Mixed Integer Programming Model

In this section we present our combined HVAC control and meeting scheduling formulation. In the description of the model, values of constants are given between square brackets. We present key elements of the HVAC model and discuss two updates over Lim et al. [15].

Following Goyal et al. [9, 10], we focus on commercial buildings with VAV-based HVAC systems, which serve over 30% of the commercial building floor

space in the United States [6]. In particular, we focus on control strategies that can be applied to each VAV box. Specifically, we determine the supply air temperature and the air flow rate, which are manipulated by a heating coil and dampers inside the VAV box. A schematic of a VAV-based HVAC system with two VAV boxes is shown in Figure 1.

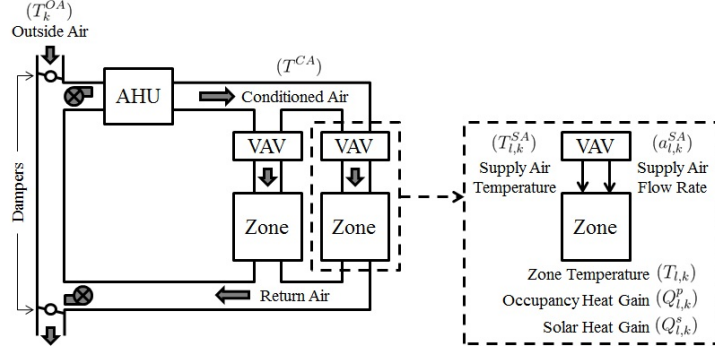


Fig. 1: VAV-based HVAC system.

Conditioned air is the air supplied by the air handling unit (AHU) to the VAV boxes. Energy is consumed by the AHU when mixing outdoor air with return air and cooling it to a pre-set conditioned air temperature T^{CA} [12.8 °C]. The AHU consumes less energy when the outdoor air temperature T^{OA} is closer to T^{CA} . Energy is also consumed by the VAV box when regulating the supply air temperature T^{SA} and the supply air flow rate a^{SA} to keep the zone temperature T within comfort bounds. In particular, the supply fan at the AHU tries to maintain a constant air pressure through the supply duct and thus may speed up or slow down depending on air flow rates used by the VAV boxes. Moreover, in order to keep the zone temperature within comfort bounds the conditioned air may need to be reheated in the VAV box.

3.1 Objective Function

We want to schedule meetings in such a way that the HVAC energy consumption is minimized. Thus, the objective function is defined as minimizing the energy use of fan, air-conditioning and heating operations. Let $K = 1, \dots, n$ be an ordered set of time steps, where the length of each time step is $t_k - t_{k-1} = \Delta t$. Let L be the set of zones in the building. We assume that each zone corresponds to one room. In general, however, a zone may correspond to a set of rooms. The objective function is defined as

$$\text{Min} \sum_{k \in K} \left(p_k^{fan} + p_k^{cond} + \sum_{l \in L} p_{l,k}^{heat} \right) \times \Delta t \quad (1)$$

where

$$p_k^{fan} = \beta \sum_{l \in L} a_{l,k}^{SA} \quad \forall k \in K \quad (2)$$

$$p_k^{cond} = C^{pa} (T_k^{OA} - T^{CA}) \sum_{l \in L} a_{l,k}^{SA} \quad \forall k \in K \quad (3)$$

$$p_{l,k}^{heat} = C^{pa} (T_{l,k}^{SA} - T^{CA}) a_{l,k}^{SA} \quad \forall l \in L, k \in K \quad (4)$$

Constraints (2)-(4) determine the values of the variables p_k^{fan} , p_k^{cond} , $p_{l,k}^{heat}$, which respectively correspond to the energy consumed by the AHU for air conditioning, the supply fan for maintaining air pressure, and the VAV box for reheating the air. It is assumed that a forecast of the outdoor temperature T_k^{OA} is available for each time step k . Also, β is the fan power coefficient [0.65] and C^{pa} is the heat capacity of air at constant pressure [1.005 kJ/kg·K].

3.2 HVAC Model

We want to determine the air flow rate $a_{l,k}^{SA}$ and supply air temperature $T_{l,k}^{SA}$ for each zone $l \in L$ and each time step $k \in K$ so as to minimize total energy consumption. In order to keep zone temperature within comfort bounds, we introduce the variables $T_{l,k}$ to represent the indoor temperature in zone l at time step k . We also introduce the variable $z_{l,k}$, which is equal to 1 if zone l is occupied at time step k and equal to 0 otherwise. The meaning of $z_{l,k}$ will become clearer when we describe the scheduling constraints in section 3.3. The variables below are used in the HVAC model to determine how much energy is needed to, for example, cool a zone from say 28 °C (82.4 °F) to 23 °C (73.4 °F).

$$T^{unocc,lb} + \delta^{lb} z_{l,k} \leq T_{l,k} \leq T^{unocc,ub} - \delta^{ub} z_{l,k} \quad (5)$$

$$T^{CA} \leq T_{l,k}^{SA} \leq T^{SA,ub} \quad \forall l \in L, k \in K \quad (6)$$

$$a^{SA,lb} \leq a_{l,k}^{SA} \leq a^{SA,ub} \quad \forall l \in L, k \in K \quad (7)$$

Constraints (5) describe the comfort bounds on zone temperature. When a zone is occupied, the zone temperature must be within a specified comfort interval $(T^{occ,lb}, T^{occ,ub})$ [(21 °C, 23 °C)]. At other times the zone temperature can fluctuate more freely $(T^{unocc,lb}, T^{unocc,ub})$ [(16 °C, 28 °C)]. These bounds can be set to reflect individual building guidelines. We use $T^{occ,lb} = T^{unocc,lb} + \delta^{lb}$ and $T^{occ,ub} = T^{unocc,ub} - \delta^{ub}$, where δ^{lb} and δ^{ub} are appropriately valued constants. Constraints (6) and (7) ensure that the supply air temperature and the air flow rate are bounded by the HVAC operational capacity. The supply air temperature $T_{l,k}^{SA}$ may supply conditioned air at temperature T^{CA} , or heat the conditioned air up to $T^{SA,ub}$ [40 °C]. The air flow rate can fluctuate between $a^{SA,lb}$ [0.108 kg/s] and $a^{SA,ub}$ [5.0 kg/s], where the lower bound is determined by the ASHRAE ventilation standard and the upper bound is reached when the dampers are fully open.

The most involved constraints in the HVAC model come from modeling building thermal dynamics. We use a lumped resistor-capacitor (RC) network, which is commonly used in [7–9] for constructing a model of the transient heat flow through solid surfaces, such as walls and windows. To predict zone temperatures we use the following parameters as inputs: (1) characteristics of the supply air (flow rate and temperature), (2) thermal heat gain due to occupants, (3) thermal heat gain due to solar radiation, and (4) outdoor temperature.

In Lim et al. [15], the temperature in a zone is affected by both internal and external walls. In our experiments, however, we observed that the temperature in a zone is mostly affected by the HVAC, occupants in the zone, and the heat flow through external walls. The heat transfer from neighboring rooms through the internal walls is negligible based on our building settings. Hence, in this model we only consider external walls, that is, those with an outside facing.

The temperature constraints are presented below.

$$\begin{aligned} \begin{bmatrix} \dot{T}_{l,k} \\ \dot{T}_{l,k}^1 \\ \vdots \\ \dot{T}_{l,k}^N \end{bmatrix} &= \begin{bmatrix} -\frac{1}{C_l}(\sum_{n=1}^N \frac{1}{R_l^n} + \frac{1}{R_l^w}) & \frac{1}{C_l R_l^1} & \cdots & \frac{1}{C_l R_l^N} \\ \frac{1}{C_l^1 R_l^1} & -\frac{1}{C_l^1}(\frac{1}{R_l^1} + \frac{1}{R_l^1}) & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{C_l^N R_l^N} & 0 & 0 & 0 - \frac{1}{C_l^N}(\frac{1}{R_l^N} + \frac{1}{R_l^N}) \end{bmatrix} \begin{bmatrix} T_{l,k-1} \\ T_{l,k-1}^1 \\ \vdots \\ T_{l,k-1}^N \end{bmatrix} \\ &+ \begin{bmatrix} \frac{1}{C_l R_l^w} & 0 & \frac{1}{C_l^t} \\ \frac{1}{C_l^1 R_l^1} & \frac{1}{C_l^1} & 0 \\ \vdots & \vdots & \vdots \\ \frac{1}{C_l^N R_l^N} & \frac{1}{C_l^N} & 0 \end{bmatrix} \begin{bmatrix} T_{k-1}^{OA} \\ Q_{l,k-1}^s \\ Q_{l,k-1}^p \end{bmatrix} + \begin{bmatrix} \frac{\Delta H_{k-1}^l}{C_l^t} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \end{aligned} \quad (8)$$

$$Q_{l,k}^p = q^p pp_{l,k} \quad \forall l \in L, k \in K \quad (9)$$

$$\Delta H_{l,k} = C_l^{p,a} a_{l,k}^{SA} (T_{l,k}^{SA} - T_{l,k}) \quad \forall l \in L, k \in K \quad (10)$$

Constraints (8) model the temperature dynamics in zone l when considering a room with N external walls. The expression $\dot{T}_{l,k} = \frac{T_{l,k} - T_{l,k-1}}{\Delta t}$ is the rate of change in zone temperature and $\dot{T}_{l,k}^n$ is the rate of change in temperature of the external wall(s) $n = 1, \dots, N$. C_l and C_l^n respectively denote the thermal capacitance of zone l and of the external wall n in zone l . R_l^n and R_l^w respectively represent the thermal resistance of wall n and the windows separating zone l with the outdoors. Constraints (9) determine the internal heat gain due to occupants $Q_{l,k}^p$, which is calculated by multiplying the heat energy q^p [75W] with the number of occupants $pp_{l,k}$. The value of the variable $pp_{l,k}$ is determined in the scheduling model, which is described in the next section. Constraints (10) determine the enthalpy of the heat supplied and extracted by the HVAC.

Note, constraints (4) and (10) contain bilinear terms $a_{l,k}^{SA} T_{l,k}^{SA}$ and $a_{l,k}^{SA} T_{l,k}$, which turns the problem into a mixed integer non-linear programming (MINLP) problem. We use the reformulation method of McCormick [18], which relaxes the original nonlinear equality constraints into a set of linear inequality constraints.

3.3 Scheduling model

Let M be a set of meetings. Each meeting $m \in M$ has a duration τ_m , a set of allowable start time steps $K_m \subseteq K$, a set of allowable zones $L_m \subseteq L$ and a list of attendees $P_m \subseteq A$, where A is the set of all attendees. The sets K_m and L_m can be used to express a number of constraints, such as, room capacity requirements, room equipment requirements, and time restrictions. Moreover, in order to express meeting conflict constraints due to attendees we let $N \subseteq 2^M$ be the set of meeting sets that have at least one attendee in common, that is $N = \{M_i \subseteq M \mid \forall m, m' \in M_i, P_m \cap P_{m'} \neq \emptyset\}$.

We want to determine $x_{m,l,k}$, which is equal to 1 if meeting m is scheduled to start at time step k in zone l and equal to 0 otherwise. The scheduling model interacts with the HVAC model via the variables $z_{l,k}$ that indicate whether zone l is occupied at time step k or not, and via the variables $pp_{l,k}$ that indicates the number of people in zone l at time step k .

The constraints are as follows.

$$\sum_{l \in L_m, k \in K_m} x_{m,l,k} = 1 \quad \forall m \in M \quad (11)$$

$$\sum_{\substack{m \in M, k' \in K_m: \\ l \in L_m, k - \tau_m + 1 \leq k' \leq k}} x_{m,l,k'} \leq z_{l,k} \quad \forall l \in L, k \in K \quad (12)$$

$$\sum_{\substack{m \in M, k' \in K_m: \\ l \in L_m, k - \tau_m + 1 \leq k' \leq k}} x_{m,l,k'} \times |P_m| = pp_{l,k} \quad \forall l \in L, k \in K \quad (13)$$

$$\sum_{\substack{m \in \nu, l \in L_m, k' \in K_m: \\ k - \tau_m + 1 \leq k' \leq k}} x_{m,l,k'} \leq 1 \quad \forall k \in K, \nu \in N \quad (14)$$

Constraints (11) require that all meetings must be scheduled exactly once. Constraints (12) state that no more than one meeting can occupy a zone at any time. Observe that the right hand side in this constraint is either zero or one, which limits the number of meetings to at most one. Also, when the left hand side equals one then the zone must be occupied. Constraints (13) determine the number of occupants in a zone $pp_{l,k}$, which is a variable used in the HVAC model and finally constraints (14) ensure that meetings with at least one attendee in common cannot be scheduled in parallel.

One issue with the current model (and the one used in [15]) is that it can have a large number of equivalent solutions when two or more meetings are identical. For example, let meetings 1 and 2 have the same time windows, same number of attendees, and same meeting conflicts. In this case, a solution in which $x_{1,l,k} = 1$ and $x_{2,l,k} = 0$ would be equivalent to one in which $x_{1,l,k} = 0$ and $x_{2,l,k} = 1$. In order to avoid the computational cost of generating both solutions we reduce the number of integer variables by defining meeting types. Meetings that are identical are considered to be of the same meeting type. Since the number of meeting types is smaller than the number of meetings, it allows for a more simplified model

that reduces symmetry. Without changing the model in its entirety we simply redefine M to be the set of meeting requests and replace constraint (11) with the one below to state that all meeting types must be scheduled ψ_m times, where ψ_m represents the number of meetings of type m .

$$\sum_{l \in L_m, k \in K_m} x_{m,l,k} = \psi_m \quad \forall m \in M \quad (15)$$

4 Large Neighborhood Search

LNS is a local search metaheuristic, which was originally proposed by Shaw [24]. In LNS, an initial solution is improved iteratively by alternating between a destroy and a repair step. The main idea behind LNS is that a large neighborhood allows the heuristic to easily navigate through the solution space even when the problem is highly-constrained. As opposed to a small neighborhood, which may make escaping a local minimum much harder.

An important decision in the destroy step is determining the amount of destruction. If too little is destroyed the effect of a large neighborhood is lost, but if too much is destroyed then the approach turns into repeated re-optimization. As for the repair step, an important decision is whether the repair should be optimal or not. An optimal repair will typically be slower than a heuristic, but may potentially lead to high quality solutions in a few iterations.

Our LNS approach starts with an initial feasible solution, which is generated using a greedy heuristic. First, this heuristic finds a feasible meeting schedule by minimizing the number of rooms. Second, it determines the HVAC control settings of supply air temperature and supply air flow rate to minimize energy consumption given a fixed schedule. This two-stage approach allows us to come up with an initial solution in reasonable time.

4.1 Destroy and Repair

Our LNS approach considers a neighborhood that contains a subset of the rooms or zones. In particular, we destroy the schedule in two to four randomly selected rooms. This forms a subproblem that can be solved effectively using MIP. When destroying meetings in more than four zones, MIP performance can degrade very quickly and even solving the linear programming relaxation can become quite time consuming. The repair consists of solving an energy aware meeting scheduling problem that is much smaller than the original problem. We do, however, limit MIP runtime to avoid excessive search during a repair step, and to avoid any convergence issues of the MIP problem. Setting a limit on runtime means that we do not necessarily solve the subproblem to optimality, but given that MIP solvers are anytime algorithms, we do improve solution quality in many of the LNS iterations. If we find an improved solution, then the new schedule and control settings are accepted. Otherwise, we maintain the solution that was just destroyed. Given that the LNS starts with a feasible solution and does not accept

infeasible solutions, the solution remains feasible throughout the execution of the algorithm.

We should note that we have experimented with a variety of neighborhoods. These include: destroying all meetings in randomly selected time steps, a combination of destroying all meetings in randomly selected rooms and time steps, and simply destroying a set of randomly selected meetings. Our observation was that none of these neighborhoods performed as well as destroying all meetings in a number of randomly selected rooms. Destroying selected rooms means that meetings could be rescheduled at any time during the day. This allows the model to optimize supply air flow rate and supply air temperature over all the time steps. Destroying selected time steps means that meetings may switch rooms, but may need to be scheduled to the same time step due to time window restrictions. This limits the optimization of supply air flow rate and supply air temperature due to the HVAC control constraints on neighboring time steps.

4.2 LNS Parameter Tuning

The parameters that govern the behavior of the LNS heuristic are parameters determining the number (2, 3, or 4) of rooms to destroy and the MIP runtime limit for the repair step. The probabilities on the number of rooms to destroy are defined as a 3-tuple with values ranging between $[0,1]$ and the MIP runtime limit is a parameter with values ranging between 1 and 10 seconds.

While it is possible to reason about certain parameters and their impact on overall performance, there are numerous values that these parameters can take on. Even though we consider only 4 parameters, it is impossible to try all possible configurations because of their continuous domains. Note, even with discretized domains with reasonable level of granularity it remains impractical to try out all configurations. As a result, we use the automated algorithm-configuration method called Sequential Model-based Algorithm Configuration (SMAC) [11] to optimize these parameters.

SMAC can be used to train parameters in order to minimize solution runtime, or to optimize solution quality. In our case, we fix the runtime and minimize energy consumption. We generate problem instances with different degrees of constrainedness and train the parameters to achieve the average best quality for all input scenarios.

Given a list of training instances and corresponding feature vectors, SMAC learns a joint model that predicts the solution quality for combinations of parameter configurations and instance features. These information are useful in selecting promising configurations in large configuration spaces. For each training instance we computed up to 17 features, including: (1) number of constraints, (2) number of variables, (3) number of non-zero coefficients, (4) number of meetings, (5) number of meeting types, (6) scheduling flexibility, (7) average duration of meetings, (8) number of meeting slots per day, (9) total number of meeting slots, (10)-(14) number of rooms in up to 5 building types, and (15)-(17) minimum, maximum, and average difference between outdoor temperature and temperature comfort bounds. These features reflect problem characteristics and are used

by SMAC to estimate performance across instances and generate a set of new configurations.

Given a list of promising parameter configurations, SMAC compares them to the current incumbent configuration until a time limit is reached. Each time a promising configuration is compared to the incumbent configuration, SMAC runs several problem instances until it decides that the promising configuration is empirically worse or at least as good as the incumbent configuration. In the latter case the incumbent is updated. In the end, the configuration selected by SMAC is generalized to all problem instances in the training set.

5 Experimental Results

Before describing our main results, we first point out the typical behavior of solving energy aware meeting scheduling as a MIP. Figure 2 shows the performance of the MIP approach on two typical problem instances when given 2 hours of runtime. In general, MIP convergence on larger problems is slow and sometimes MIP fails to converge even after 2 hours. This is exactly why we developed the LNS approach. The typical performance of the LNS approach is also given in Figure 2. In the figure, LNS was given only 15 minutes of runtime but it is capable of returning significantly better results when compared to MIP.

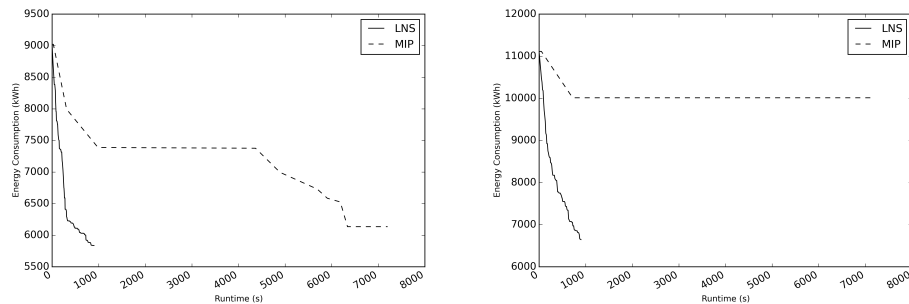


Fig. 2: Typical performance of MIP (2 hours) and LNS (15 minutes) on two benchmark instances.

We analyze our LNS approach by considering 8 problem sets. Each problem set contains 10 problem instances that we built by adding energy related information to instances extracted from the PATAT timetabling dataset [19]. The problem sets differ by the number of meetings (M) and the number of rooms (R). Specifically, our problem sets are referred to as 20M-20R, 50M-20R, 100M-20R, 200M-20R, 50M-50R, 100M-50R, 200M-50R, and 500M-50R, where 20M-20R represents the problem set with 20 meetings and 20 rooms. All our experiments were run on a cluster that consists of a $2 \times$ AMD 6-Core Opteron 4334, 3.1GHz with 64GB memory.

Building Types	External Wall		Internal Wall		Window
	(TR)	(TC)	(TR)	(TC)	
1	3	120	1.5	120	0.5
2	3	140	1.5	140	0.5
3	3	240	1.5	240	0.5
4	6	120	3	120	0.5
5	6	240	3	240	0.5

Table 1: Total thermal resistance (TR) $\left(\frac{m^2K}{W}\right)$ and thermal capacitance (TC) $\left(\frac{KJ}{m^2K}\right)$ of the walls and the window for five types of zones. R and C of each wall can be derived by dividing TR and multiplying TC with area size respectively.

In each problem set we must schedule up to 500 meetings whose durations are 1 or 1.5 hours. The meetings must be scheduled over a period of 5 summer days. The available rooms are located in 5 buildings, that differ by their thermal resistance and capacitance as specified in Table 1. We use a 1×4 zone layout where each zone has the same thermal resistance and capacitance as its neighboring zones. Moreover, all rooms have the same geometric area of $6 \times 10 \times 3 \text{ m}^3$ with a window surface area of $4 \times 2 \text{ m}^2$ and a capacity of 30 people. The solar gain ranges from 50 to 350 W/m^2 during the day. All meetings have between 2 and 30 attendees and we vary the scheduling flexibility for each meeting with an allowable time range of one or two random days (between 09:00-17:00) within the 5 summer days.

First, we used SMAC to tune the parameters for all 80 instances. However, it is possible that one set of parameter configurations might not produce best results across all problem sets. For example, Malitsky et al. [17] observed that instance specific algorithm configuration finds good quality solutions for large sized instances in limited time. Hence, second we independently tuned the parameters for the 8 problem sets. In each case, we used 0.33, 0.33, and 0.34 as the default destroy probabilities and 5 seconds for the default MIP runtime during each repair step. SMAC trains on 60% of the instances and cross-validates with the remaining 40%.

On average, SMAC generated 300 configurations for each problem set. Figure 3 shows the result of the best parameter configurations as determined by SMAC for all 80 instances (AllM-AllR), and for the each of the 8 problem sets. In the end, a MIP runtime of 8.5 seconds, and probabilities of 0.64, 0.28, 0.08 to destroy 2, 3, and 4 rooms respectively were determined to be the best settings by SMAC for all 80 instances. However, the best parameter configurations do vary somewhat for the different problem sets.

Figure 4 shows the performance improvement of MIP and LNS compared to the initial feasible solution, which is referred to as the heuristic solution (HS), over 500 runs. For each run, both the MIP and LNS approach were seeded with HS as an initial solution. Both MIP and LNS were given the same runtime limit of 15 minutes and HS was given 60 seconds. LNS was executed using the parameter configurations that were determined best for each problem set. The results show that LNS significantly improves over MIP when given limited runtime. Overall,

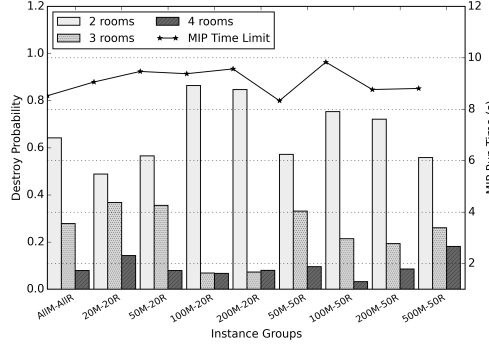


Fig. 3: Parameter configurations as determined by SMAC.

the improvement of LNS over MIP is between 14% and 36%. Note that in the largest problem instances, those in 500M-50R, MIP often fails to find even a slight improvement over the given initial solution HS in 15 minutes.

We reran all problem instances in Figure 4 using the AllM-AllR parameter settings. In this case, the performance of LNS decreased by 1 to 2%. Hence, instance specific algorithm configuration did have some impact, but even without it LNS performed significantly better than MIP.

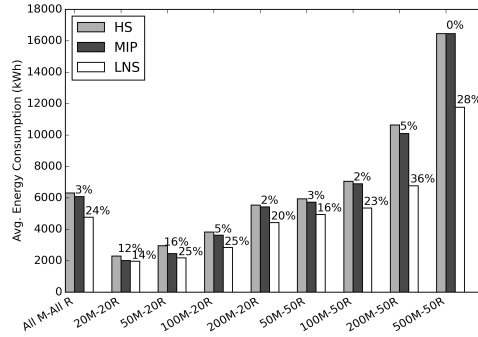


Fig. 4: Performance improvement of MIP and LNS over heuristic solution (HS). MIP and LNS runtime 15 minutes.

Figure 5 provides insight into the optimality gap of the MIP approach and in the variation in solution quality. The MIP optimality gap after 15 minutes runtime is shown on the left. Each box shows the median and the upper and lower quantiles of the optimality gap. The endpoints indicate the maximum and minimum. As observed previously the MIP approach fails to converge on the larger problem instances. Moreover, as can be seen from the figure, MIP's

performance substantially degrades as problems become more constrained and exhibit a higher number of meetings to number of rooms ratio.

The variation in solution quality is shown on the right. The median of LNS always falls below the lower quantile of MIP. We note, however, that for problem instances in 20M-20R, MIP and LNS have almost similar performance. For even smaller problem instances MIP tends to be very effective, which is exactly why we have combined the two. We believe that the combination of LNS and MIP is especially good when considering highly constrained problems. LNS is used to destroy and repair the solution space and MIP is used to find good local solution, possibly optimal solution, in a short amount of time.

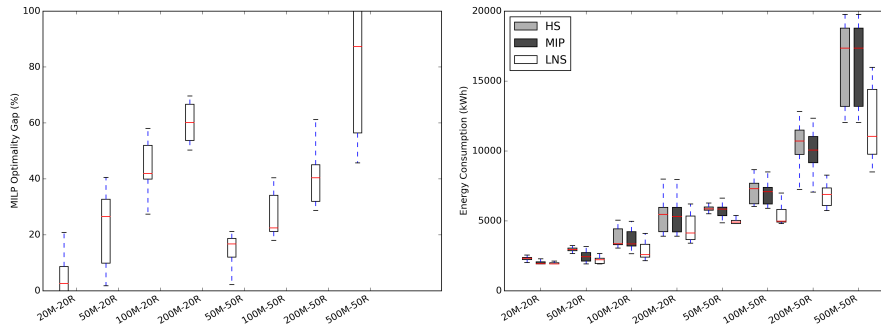


Fig. 5: MIP optimality gap (left) and HS/LNS/MIP solution values (right)

6 Conclusions

In this paper we extend the work by Lim et al. [15], which introduced a MIP model for energy aware meeting scheduling. This previous work shows that combining HVAC control with meeting scheduling can lead to substantial improvements in energy efficiency. The MIP model that is described, however, only solves problem instances that involve a small number of meetings and rooms. We combine MIP with LNS in order to scale to larger problems.

We developed a heuristic to generate an initial feasible solution quickly, which we use to warm start both the MIP and LNS approach. In our experiments, the most effective neighborhood was one that destroys and repairs all meetings scheduled in 2 to 4 rooms. The resulting subproblem was small enough for MIP to solve to (near) optimality and, at the same time, large enough to explore alternate solutions. We studied the performance of MIP and LNS and demonstrated the potential of our LNS approach for effectively tackling large-scale HVAC control and meeting scheduling problems. The LNS achieves 14 to 36% better energy savings than the MIP approach when both given a runtime of 15 minutes. In order to provide an absolute sense of the solution quality, we plan to evaluate our schedules with the EnergyPlus simulator.

We are interested in exploring new algorithmic approaches that allows us to scale even further. We are particularly interested in investigating symmetry

breaking in MIP. Symmetry leads to a large number of equivalent solutions, which causes branch-and-bound to be ineffective. While we dealt with symmetries due to meetings with similar characteristics by introducing meeting types, symmetries still exist in our MIP formulation due to rooms with similar characteristics. Introducing room types, however, may not be possible because room temperature at time step t is dependent on time step $t - 1$. In future work, we aim to identify branching strategies that can better deal with symmetries in our MIP formulation.

Moreover, we are also interested in investigating an online stochastic approach to our HVAC control and meeting scheduling problem. Such an approach can deal with current requests, future requests, changes and cancellation of requests, but also with the uncertainty around outdoor air temperature and weather conditions.

Acknowledgments

Thanks to Pascal Van Hentenryck for pointing us to optimization problems in the smart buildings space and to Philip Kilby for helpful discussions on LNS. This work is supported by NICTA's Optimisation Research Group as part of the Future Energy Systems project. NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program.

References

1. ASHRAE: ASHRAE handbook: Fundamentals. American Society of Heating, Refrigerating and Air-Conditioning Engineers (2013)
2. Bloomfield, D., Fisk, D.: The optimisation of intermittent heating. *Building and Environment* 12(1), 43–55 (1977)
3. Crawley, D.B., Pedersen, C.O., Lawrie, L.K., Winkelmann, F.C.: Energyplus: Energy simulation program. *ASHRAE Journal* 42, 49–56 (2000)
4. Danna, E., Perron, L.: Structured vs. unstructured large neighborhood search: A case study on job-shop scheduling problems with earliness and tardiness costs. In: *Proc. International Conference on the Principles and Practice of Constraint Programming (CP)*. pp. 817–821 (2003)
5. Di Gaspero, L., Rendl, A., Urli, T.: Constraint-based approaches for balancing bike sharing systems. In: *Proc. International Conference on the Principles and Practice of Constraint Programming (CP)*. pp. 758–773 (2013)
6. EIA: US Department of Energy, CBECS detailed tables (2003), <http://www.eia.gov/consumption/commercial/>
7. Gouda, M., Danaher, S., Underwood, C.: Low-order model for the simulation of a building and its heating system. *Building Services Engineering Research and Technology* 21(3), 199–208 (2000)
8. Gouda, M., Danaher, S., Underwood, C.: Building thermal model reduction using nonlinear constrained optimization. *Building and Environment* 37(12), 1255–1265 (2002)

9. Goyal, S., Barooah, P.: A method for model-reduction of non-linear thermal dynamics of multi-zone buildings. *Energy and Buildings* 47, 332–340 (2012)
10. Goyal, S., Ingley, H.A., Barooah, P.: Occupancy-based zone-climate control for energy-efficient buildings: Complexity vs. performance. *Applied Energy* 106, 209–221 (2013)
11. Hutter, F., Hoos, H.H., Leyton-Brown, K.: Sequential model-based optimization for general algorithm configuration. In: *Proc. International Conference on Learning and Intelligent Optimization (LION)*. pp. 507–523 (2011)
12. Kwak, J.y., Kar, D., Haskell, W., Varakantham, P., Tambe, M.: Building thinc: User incentivization and meeting rescheduling for energy savings. In: *Proc. International Conference on Autonomous Agents and Multi-agent Systems (AAMAS)*. pp. 925–932 (2014)
13. Kwak, J.y., Varakantham, P., Maheswaran, R., Chang, Y.H., Tambe, M., Becerik-Gerber, B., Wood, W.: Tesla: An energy-saving agent that leverages schedule flexibility. In: *Proc. International Conference on Autonomous Agents and Multi-agent Systems (AAMAS)*. pp. 965–972 (2013)
14. LeBras, R., Dilkina, B.N., Xue, Y., Gomes, C.P., McKelvey, K.S., Schwartz, M.K., Montgomery, C.A.: Robust network design for multispecies conservation. In: *Proc. AAAI Conference on Artificial Intelligence (AAAI)*. pp. 1305–1312 (2013)
15. Lim, B.P., van den Briel, M., Thiébaux, S., Backhaus, S., Bent, R.: Hvac-aware occupancy scheduling. In: *Proc. AAAI Conference on Artificial Intelligence (AAAI)* (2015)
16. Majumdar, A., Albonesi, D.H., Bose, P.: Energy-aware meeting scheduling algorithms for smart buildings. In: *Proc. ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings (BuildSys)*. pp. 161–168 (2012)
17. Malitsky, Y., Mehta, D., OSullivan, B., Simonis, H.: Tuning parameters of large neighborhood search for the machine reassignment problem. In: *Proc. Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR)*. pp. 176–192 (2013)
18. McCormick, G.P.: Computability of global solutions to factorable nonconvex programs: Part I- convex underestimating problems. *Mathematical programming* 10(1), 147–175 (1976)
19. Melbourne University: Patat dataset (2002), <http://goo.gl/XtNwpR>
20. Oldewurtel, F., Parisio, A., Jones, C.N., Gyalistras, D., Gwerder, M., Stauch, V., Lehmann, B., Morari, M.: Use of model predictive control and weather forecasts for energy efficient building climate control. *Energy and Buildings* 45, 15–27 (2012)
21. Pan, D., Yuan, Y., Wang, D., Xu, X., Peng, Y., Peng, X., Wan, P.J.: Thermal inertia: Towards an energy conservation room management system. In: *Proc. IEEE International Conference on Computer Communications (INFOCOM)*. pp. 2606–2610 (2012)
22. Portland State University: Efficient class scheduling conserves energy (2012), <http://goo.gl/cZwgB>
23. Rendl, A., Prandtstetter, M., Hiermann, G., Puchinger, J., Raidl, G.: Hybrid heuristics for multimodal homecare scheduling. In: *Proc. Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR)*. pp. 339–355 (2012)
24. Shaw, P.: Using constraint programming and local search methods to solve vehicle routing problems. In: *Proc. International Conference on Principles and Practice of Constraint Programming (CP)*. pp. 417–431 (1998)