Multi-Agent Stochastic Simulation of Occupants for Building Simulation

Jacob Chapman[1], Peer-Olaf Siebers[1], Darren Robinson[2]
[1]School of Computer Science, The University of Nottingham, UK
[2]Sheffield School of Architecture (SSoA), University of Sheffield, UK

Abstract

This paper introduces a new general platform for the simulation of occupants' presence and behaviours. Called No-MASS (Nottingham Multi-Agent Stochastic Simulation platform) the platform takes a selection of well validated stochastic models to generate a synthetic population of agents, predicts their presence and, in the case of residences also their activities and inferred locations, as well as their use of windows, lights and blinds. A social interaction framework is used to emulate negotiations amongst the members of diverse populations. Furthermore, machine learning techniques allow the agents to learn dynamic behaviours that maximise energy and/ or comfort rewards. This is complemented by a belief-desire-intent framework for the representation of less sophisticated behaviours for which data is scarce. Using the Functional Mockup Interface (FMI) co-simulation standard No-MASS is coupled with EnergyPlus: EnergyPlus parses environmental parameters to No-MASS which in turns parses back the energetic consequences of agents behaviours. Simulations demonstrating the range of results that No-MASS can produce are undertaken and presented.

Introduction

Although a powerful building (re-)design decision support tool, building performance simulations can be subject to limitations. Studies have found that buildings may use twice as much energy as predicted at the design stage. Bordass et al. (2001) studied 16 buildings which were predicted to have low energy use, however real world tests showed they were not low energy, but used as much as an average building. It has been reported that for high energy buildings such as labs there is on average a factor of 2.5 difference between predicted and actual energy use (Turner and Frankel, 2008). Baker and Steemers (2003) similarly suggested that occupants are responsible for a factor of 2.5 variation in energy use amongst otherwise identical buildings. Even under the stringent design and construction standards imposed by Passivhaus, Blight and Coley (2013) observed on average a difference of 21% between simulated and actual energy use. It is important that the energetic consequences of occupants behaviours be rigorously and systematically taken into account in building performance simulation. An effective framework for simulating occupants in building performance simulation is multi-agent simulation, facilitating the use of a rich set of complementary modelling tools including: stochastic models, theory driven rules, social interactions and machine learning techniques.

Multi-agent simulation is a tool that has been developed primarily in the social sciences to effectively model human interaction (Bonabeau, 2002; Zhang et al., 2011). Its use in the social sciences has typically been to study behaviours that emerge from bottom up interactions, allowing the creator to make judgements as to what has caused these emergent behaviours and whether they correspond with expectation from social theory. An agent should have the following properties: they should be autonomous, have social ability, perceive and react to the environment and be proactive with their choices (Wooldridge and Jennings, 1995). Each agent has rules and behaviours, making them excellent in principle at modelling group and individual interactions (Axtell, 2000).

In this paper we discuss our Nottingham Multi-Agent Stochastic Simulation (No-MASS) framework. No-MASS integrates existing data-driven stochastic models of occupant interaction into a tool that can be coupled with building or urban performance simulation tools. This coupling allows for simulated occupants to make changes within the simulated building environment and to receive responses arising from the effects of their interactions which may stimulate future interactions. But current data-driven stochastic models do not cover all of the energy related behaviours of occupants. Therefore a belief-desire-intent (BDI) rule system is used to model straightforward interactions for which data is scarce. For example switching off the light when going to sleep, closing windows whilst bathing or opening curtains and windows upon awakening. Agents have unique desires and acting on them may cause changes within the environment that are in conflict with the desires of other agents. To solve this problem an agent social interaction model is developed to emulated negotiated interactions between agents. For more complex interactions where BDI rules would be difficult to design, agent machine learning techniques are used, allowing the agent to learn how to respond to different stimuli to maximise a (e.g. energy or comfort) reward

https://doi.org/10.26868/25222708.2017.051

function. In what follows we explain the principles of these different multi-agent modelling strategies and illustrate their application.

No-MASS

A conceptual flow diagram of No-MASS is depicted in figure 1. First No-MASS builds an agent population, assigning a profile to each member, dependent on the input parameters supplied in the No-MASS configuration file. These profiles influence the later models predicting activities (e.g. sleeping, bathing, watching TV) and dependent behaviours (e.g. opening windows or lowering shading devices).

Once the agent profiles are assigned No-MASS proceeds according to one of two scenarios. If the building is non-residential, chains of presence and absence are calculated (Page et al., 2008); otherwise for residential buildings chains of activities are calculated (Jaboob, 2015), and the corresponding activity locations are assigned to each activity. Communication with the Building Performance Simulation (BPS) tool is then performed. The BPS that we have coupled with No-MASS is EnergyPlus, however No-MASS utilises the generic Functional Mockup Interface (FMI) co-simulation standard, so that it can also be coupled with other FMI compliant BPS software. The BPS calculates the environmental conditions within the zones that the agents have been allocated to, these are then parsed to No-MASS via the FMI. Each agent then retrieves its pre-processed location or presence. State parameters are then set based on the activity that is performed, affecting the agents clothing level, metabolic rate etc. We then calculate the metabolic gains of each individual occupant. Next we call models predicting the agents use of shades (Haldi and Robinson, 2010), windows (Haldi and Robinson, 2009) and lights (Reinhart, 2004)

These initial core models provide a solid foundation for simulating the impact on building performance of occupants behaviours. But they are by no means complete. As noted earlier, these data-driven models are complemented by strategies to model phenomena for which data is scarce: BDI rules and Agent-Learning; as well as a framework to emulated negotiated interactions amongst diverse members of a population, through a simple vote casting and processing mechanism. Finally the results of the interactions are parsed back to the building solver, and the process repeated until the simulation ends.

Stochastic Models

We have two methods for calculating presence within a building, the choice of which depends on the type of building. For residential buildings, presence (or rather absence) is predicted directly by the activity model (Jaboob, 2015). This model predicts the time-dependent probability that one of a set of ten activities will be performed in the home. These activities include sleeping, passive, audio/visual, IT, cooking, cleaning, metabolic, washing appliance use, personal washing and absence from the building. They are modelled as a time-dependent Bernoulli process using multinomial logistic regression. As the probabilities are only dependent on time it is possible to generate a 10 by 24 matrix giving the probability of performing each activity at a given hour; though the corresponding model can also be re-called within the hour for sub-hourly simulation timesteps, interpolating between adjacent hours. Models can also be estimated for subpopulations of the time use survey dataset from which they are derived, to give probabilities that depend for example on age, employment status, season or day of the week. Dependent on the activity, values of clothing and metabolic rate are modified and metabolic gains are calculated, based on ISO 7730 (ISO, 2005) and using the standard physical (air temperature, radiant temperature, relative air velocity and relative humidity) and personal (clothing level and metabolic rate) parameters. With the exception of an assumed relative air velocity of 0.1 m/s, the physical parameters are supplied by EnergyPlus. We also infer a location for the agents activity. For example, if the agent is in the sleeping state it can be assumed that they are in their bedroom. We are currently working on the completion of a new synthetic population generator that assigns socio-demographic characteristics to the members of households for residential simulations; but in the meantime these are user-defined in the configuration file. For the simulation of non-residential buildings a presence model (Page et al., 2008) predicts when an occupant is present within their office, based on a time inhomogeneous Markov chain, using a mobility parameter and a time-dependent profile of the probability of presence P(t) as input. We calculate metabolic heat gains in the same manner as for residences, but assuming a metabolic rate and clothing level that is appropriate to offices.

We use the model of Haldi and Robinson (2009) to predict interactions with windows. This is a hybrid model, predicting transitions in opening status using a presence dependent Markov chain and, in the cases of transitions to the open state, predicting the duration for which the window stays open using a Weibull distribution. The probability of transition in window opening state, from i to $j(i, j = 0, 1)$ is calculated using a multivariate logistic regression model. The shading action model (Haldi and Robinson, 2010) predicts lowering and raising probabilities, which are also based on Markov chains. Upon an agents arrival the first step in this model is to determine the probability with which a raising or lowering action will take place. If the shade is lowered or raised we then predict whether the shade is fully raised or lowered. If the shades are only partially raised or lowered, an unshaded fraction is drawn from a Weibull distribution. Since shading control is only represented as a

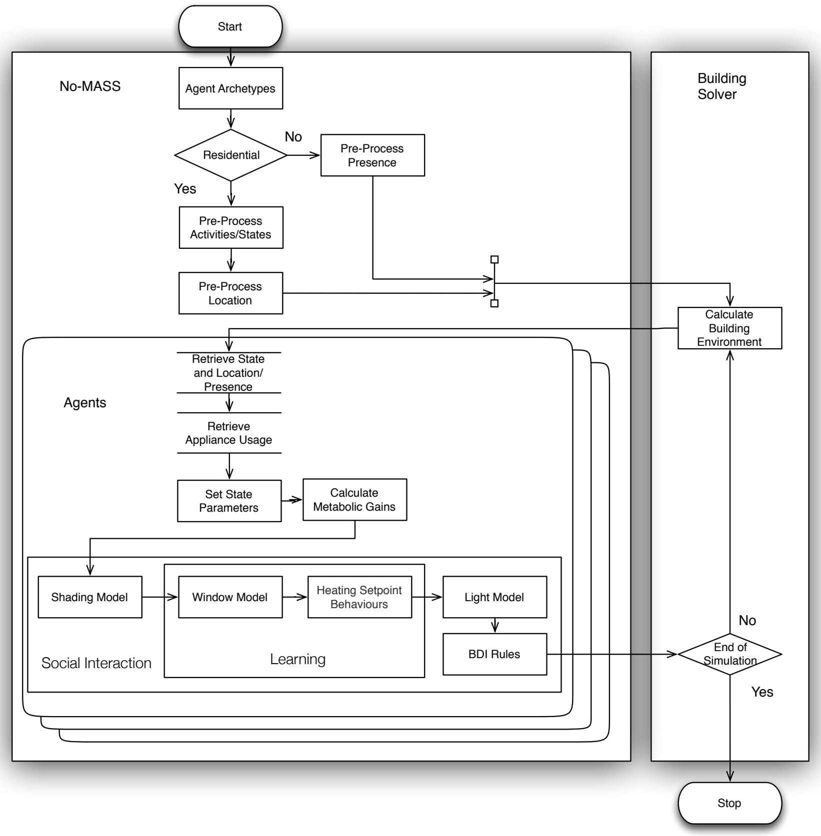


Figure 1: No-MASS Flow Diagram

Boolean operation (shades are either up or down), the EnergyPlus source code had to be altered to provide a function that moderates the radiation transmitted through the window in proportion to our unshaded fraction. This function is accessed through the Function Mockup Interface. The use of lights within No-MASS is predicted based on the Lightswitch-2002 algorithm (Reinhart, 2004). Indoor zone illuminance is taken from EnergyPlus at each time step and used to compute the probability of turning the lights on when the agent arrives or whilst they are present, and thus whether this action takes place. When all agents vacate the zone we predict whether the lights will be turned off, as a function of the anticipated duration of their absence, calculated by forward winding from the time of departure until the time of return for each agent. For absences greater than 0.5 hours and less than 12 hours the probability of turning lights off at departure are calculated (Pigg et al., 1996). For absences of below 0.5 hours we assume that lights remain on, whereas for absences exceeding 12 hours the lights are assumed to be turned off. The consequent lighting status (on-off) is set within EnergyPlus at each timestep as a lighting schedule for each zone within the building.

Theory Driven Models

Although good progress has been made in the development of rigorously formulated and validated data driven models of stochastic behaviours, there remain many gaps in our modelling capabilities, due to lack of corresponding data. But this need not preclude us from employing pragmatic rules that would help us to identify types of interactions that merit further study; or indeed whether simplified agent behaviours may in themselves suffice. To this end we introduce a belief-desire-intention (BDI) framework to test such what-if scenarios within No-MASS, extending its current features in a pragmatic way. A selection of what-if scenarios are implemented within No-MASS; here we discuss just one to demonstrate the concept.

No-MASS adopts an agent-oriented methodology and associated technique for BDI agents, as described in Kinny and Georgeff (1996) and Kinny et al. (1996). This methodology describes agents in terms of objects (a variable that has a set of attributes and methods) that exhibit encapsulation and inheritance. Encapsulation enables components to be grouped together under a single object, for example grouping multiple stochastic models into a single agent; while inheritance refers to the ability to reuse the same sections of code between different types of agents, for example to re-use the activity model, but with agent-specific coefficients. Using these features enables complex be-

haviours emerging from multiple agents to be efficiently coded. The approach is not language specific and presents a methodology rather than a specific set of rules to follow; enabling us to adapt these principles and roles for implementation in No-MASS.
First a belief set is derived, describing what the agent believes about the environment. The BDI framework within No-MASS requires that agents have knowledge of the windows, shades, lights, their current activity and time of day; these are parsed from EnergyPlus and the stochastic models. The agents are also given an understanding of the first arrival of the day and the last departure of the day, calculated by forward winding the predicted presence in a zone. Desires are handled by a Goal model, which consists of a goal set and the goal states. For example we may have as a goal to carry on our work-related tasks using a computer whilst at work. Thus upon arrival our agent turns on the corresponding appliance(s), these staying on until their departure; so that our agents leave their equipment on while they attend meetings or go for lunch. All goals are associated with a probability of occurrence that limits the number of times that an action will take place over the course of a simulation. Intents are managed by a plan model - sets of plans that are organised using plan diagrams (Kinny and Georgeff, 1996). Appliance use while at work in non-residential buildings requires a plan which is described as; while present in the office the agent has the state IT, as this sets their clothing and metabolic rate for the activity. On entering this state at the first arrival of the day the agent will turn the appliance from off to on. They then have the ability to remember this state implying that they remember they have left the appliance on for the length of the day, even when the agent leaves the office for intermittent periods. On the final departure of the day the agent will turn all their appliances off. We could of course be more sophisticated and emulate the modelling of tasks of varying intensity whilst present.
Integrating BDI rules within No-MASS in this way allows for the testing of a broad range of occupant interactions within simulation software. They can answer what-if scenarios, showing the effects of different interactions. Sensitivity analysis performed against the different plans also allows for recommendations to be made, in terms of where to focus future efforts of data collection and empirically validated stochastic model development, based on the corresponding impacts on comfort or on energy demands. This may also help to identify which behavioural practices should be encouraged by the future populations of the buildings being designed and thus simulated, to alleviate discomfort or to reduce energy use.

Social Interactions

(Wooldridge and Jennings, 1995) note that agents should be autonomous, have social ability, perceive and react to the environment and be proactive with their choices. So far No-MASS agents can perceive the environment and react to changes within it. They are autonomous but self interested, lacking in social ability. For example each agent decides independently whether they would like to open or close the window to minimise their discomfort. Although this is perfectly acceptable for singly-occupied spaces, it is not acceptable for the modelling spaces that are occupied by multiple people that, whilst diverse in their beliefs and desires and their understandings of how to achieve their desires, they can nevertheless reconcile their differences through negotiation. To address this we need a mechanism to emulate social interactions between diverse members of an agent population. In other words, if they intend to open a window they must negotiate with their peer group. One such mechanism if reconciling potentially conflicting desires is thorugh a voting system, similar to that employed by Ephrati and Rosenschein (1991), which has be demonstrated to be a quick and effective method for arriving at a consensus. Within No-MASS constraints can be placed on the actions that can be performed during a conflict; this is achieved through a biased voting system. Occupants may have differing authority to make choices about the environment. Some agents can have larger voting rights than others, we call these voting rights *power*, these are social laws (Shoham and Tennenholtz, 1992) built directly into agents representations. In a democratic regime of size n each agent would have identical power, define by 1/n; whereas in an authoritarian regime the controlling agents power would simply be 1.

If a single agent wishes to open a window and another two choose otherwise then, in a democratic regime, the preferences of these two would win; they get to veto the desires of their colleague. If we have a democratic group with equal numbers wishing to open or to close the window, we decide the outcome by simply tossing a coin (e.g. drawing a random number and assigning the window to state closed if below 0.5).

The approach works well in binary cases where the actions are on/ off, such as for the lighting model. However models such as the external shading interaction model predict an unshaded fraction. An agent has a choice of actions that they can take to either raise, lower or keep the shades as they are. Given a set of agents, one could choose to raise the shade to a percentage and the other lower the shade. In the first instance the voting mechanism can be used; agents can vote to raise, lower or do nothing to the shading device. To determine the percentage change that occurs, social laws are again enforced on the agents, removing the need for time consuming negotiation. A set of two agents choose to raise the shade from its current position but they both choose to open it to different levels. Here we impose a restriction on the agents that they must choose the average of the two. This will satisfy the agents need to raise the blind and allows the simulation to move on. Within No-MASS agents assess their personal preferences at each timestep for all the stochastic models, so that conflicts have to be resolved at each timestep. However, this methodology of processing votes does not increase simulation time significantly and provides a first approximation of agents negotiation mechanisms within buildings; the effects of which will be discussed later in this paper.

Machine Learning

As previously noted developing and rigorously validating stochastic models of occupants bahaviours is data intensive. Although a BDI framework is a useful companion to such models in the absence of data, its use should be restricted to relatively straightforward types of interaction. For more complex interactions, agent learning is a promising alternative, whereby agents learn from past experiences to take actions in the present that will affect their comfort in the future. Q-learning (Watkins and Dayan, 1992) allows agents to learn a response from a reward function, to decide on an action. This allows agents to develop an understanding of their preferences over time. In a Markovian domain an agent learns the best action in a given state, this is achieved by trying every action in a state and updating the expected reward with the actual reward for that action. This is useful in building performance simulation as the same methodology can be applied to areas where models are missing due to lack of data and where there is a clear link between an action and a driving stimulus. For example, does a chosen action cause comfort or discomfort, increase the reward if comfort is maintained and decrease if not. The Q Value at state s_t for action a_t is given by the function:

$$Q_t(s_t, a_t) = Q_t(s_t, a_t) + \alpha(r + \gamma * max(Q(s_{t+1}, a)) - Q_t(s_t, a_t))$$

Where the reward is r, the learning rate, α, is $0 < \alpha \leqslant 1$ and the discount Factor, γ is $0 < \gamma \leqslant 1$. The discount factor specifies how soon the agent cares about the reward, ie. if an agent is myopic, they care more about near term gains, otherwise the agent would prefer long term rewards. Long term rewards are set closer to 1. Q-learning requires a map from state to action. In building simulation heating setpoints (for which there is a lack of high quality longitudinal data) are time based, allowing a different setpoint to be set for the time of day. This gives the model its first constraint, the timestep intervals within building simulation. To keep the number of states small we limit the number of learnt states to the hours of the day, but not every day is the same; the heating set points for the working week may differ from those of the weekend. Rather than learn the optimal action for every day of the year, No-MASS agents learn for weekdays and weekends. The final constraint placed on the states is that heating demand is seasonal, it changes over time based on the season. To overcome this issue we set our agents to learn the best action for each month. No-MASS Q-learning states are therefore defined for the hours of a working weekday and the weekend for each month. No-MASS now has a set of states that an agent can be in at a given point in time, and a set of actions that an agent can perform at each state. This is the heating setpoint, which we constrain to be between the heating setback temperature and, if relevant, the cooling setpoint temperature. Q-learning requires a map from state to action. In building simulation heating setpoints (for which there is a lack of high quality longitudinal data) are time based, allowing a different setpoint to be set for the time of day. This gives the model its first constraint, the timestep intervals within building simulation. To keep the number of states small we limit the number of learnt states to the hours of the day, but not every day is the same; the heating set points for the working week may differ from those of the weekend. Rather than learn the optimal action for everyday of the year, No-MASS agents learn for weekdays and weekends. The final constraint placed on the states is that heating demand is seasonal, it changes over time based on the season. To overcome this issue we set our agents to learn the best action for each month. No-MASS Q-learning states are therefore the hours of a working weekday and the weekend for each month. No-MASS now has a set of states that an agent can be in at a given point in time, and a set of actions that an agent can perform at each state. This is the heating setpoint, which we constrain to be between the heating setback temperature and, if relevant, the cooling setpoint temperature.

With the states and actions defined a method of rewarding or punishing the agents when they perform an action needs to be considered. Heating set points are linked to an agents comfort, so a sensible solution is to reward an agent based on the outputs of a thermal comfort model; thus allowing them to learn the heating setpoint values that minimise discomfort at a given point in time. No-MASS currently has a mechanism to calculate the agents metabolic gains based on ISO:7730 (2005), which also provides the source code to calculate the Predicted Mean Vote (PMV) of each agent on a thermal sensation scale from -3 (cold), through 0 (neutral) to hot (3). We expected our agents to aim to be efficient and not wasteful; so that a PMV above 0 should be punished (at least whilst in heating mode). Conversely if the agent is too cold $PMV < -0.5$, the agent is punished once again for having selected a discomforting strategy, so that $-0.5 < PMV <= 0$ obtains a relatively high reward. It is also desirable to restrict heating while the agent is absent, otherwise an optimal learning policy would be to wastefully leave the heating on to maintain a temperature that is satisfactory at all times. To overcome this, agents are punished if the heating is above the setback temperature for more than an hour while the agent is not present. Thus our reward function is:

$$r = z * (1 * c + 0.1 * a - 0.1 * b - 0.1 * e) \\ + (1 - z) * (d * 0.1 + (1 - d) * -0.1)$$

a, b, c, d, e and z are binary operators where a is 1 when $pmv > 0$ and the heating setpoint equals the heating setback, b is 1 when $pmv < -0.5$, c is 1 when $pmv >= -0.5$ and $pmv <= 0$ and d is 1 when the heating setpoint equals the heating setback and the absence length > 1 hour. e is 1 when $pmv > 0$ and the heating setpoint temperature is greater than the setback temperature.

For the Q-learning equation our learning rate α is set to 0.1 as the environment is non deterministic, as suggested by Sutton and Barto (1998). The discount Factor γ is 0.1, making the agent short sighted, so that they prefer rewards that are short term, for discomfort to be reduced in the near future.

The Q-learning method we have described so far has no exploration over time; choosing a methodology of $\epsilon - greedy$ we enable the agents to explore the parameter space. An $\epsilon - greedy$ policy is the chance of random action taking place over the optimal. Sutton and Barto (1998) suggest that a value of 0.01 is slower at learning random actions taken but converges to a better policy than $\epsilon = 0.1$. Random actions are taken to ensure that as the agent learns all possible actions are tested over time; what may have been optimal at the start of the learning period may not be at the end. For the present case of learning heating setpoints we choose $\epsilon = 0.01$, meaning that 1% of the actions taking place are random.

Implementation

No-MASS was built from the ground up, with C++ chosen as the development language as it is simple to integrate with EnergyPlus, our chosen building performance simulation tool, which is also developed in C++. Using the same language allows for easy communication between the two tools. EnergyPlus is well tested and well documented, allowing us to readily understand how to connect to it. The No-MASS platform connects to EnergyPlus using the Functional Mockup Interface (FMI) (Nouidui et al., 2013), which is an open standard so that No- MASS could in principle be integrated with any other FMI compliant simulation tool. FMI specifies how an array of double precision values is parsed between EnergyPlus and No-MASS. The array of values that No-MASS receives at each time step is defined in the XML file ModelDescription.xml. At the beginning of the time step the following environmental variables are received: horizontal sky illuminance, rain status, outdoor air dry-bulb temperature, zone air temperature, zone humidity, indoor radiant temperature and indoor illuminance. Returned to EnergyPlus are the number of occupants in a zone, their metabolic gains, appliance gains, the window status, the blind unshaded fraction, the lighting status and the heating setpoint. These return values overwrite EnergyPluss values for the next timestep.

No-MASS reads in data from an XML file called No-MassConfig.xml. This file contains information about the occupants that is used to build and attribute the agent population (a series of parameters defining the socio-demographic characteristics of the agent, ie. gender, age, income level, etc), and the subsequent processing of an agent activity profile that is used to calculate the probability of an activity taking place at each timestep and the corresponding location. It also defines the window and shading model coefficients for each model, allowing for diversity between occupants in terms of their use of windows and shading devices.

To enable researchers and early adopters to use and test No-MASS, user interface components controlling No- MASS were integrated with the DesignBuilder simulation tool. DesignBuilder provides access to EnerygPlus and now No-MASS through an easy-to-use interface. In this way No-MASS can now in principle be used by any building designer without the need to know the inner workings of EnergyPlus, FMI and No-MASS.

Case Study & Results

To demonstrate the application of No-MASS and its coupling with EnergyPlus we examine a mono-zone office shown in Figure 2, located in Nottingham, UK. Results from No-MASS are compared to the results arising from standard deterministic schedules and rules for the relevant office typology (or template) used by the DesignBuilder interface.

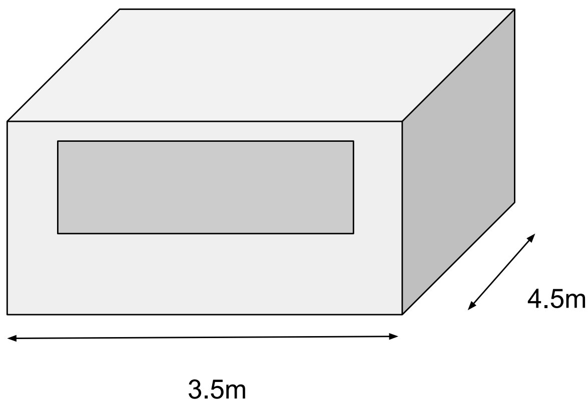


Figure 2: Office

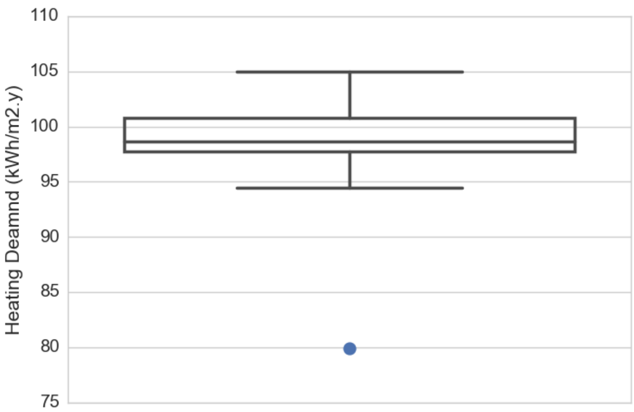


Figure 3: Simulation results for heating demand, (boxplot) 100 replicates stochastic, (circle) deterministic

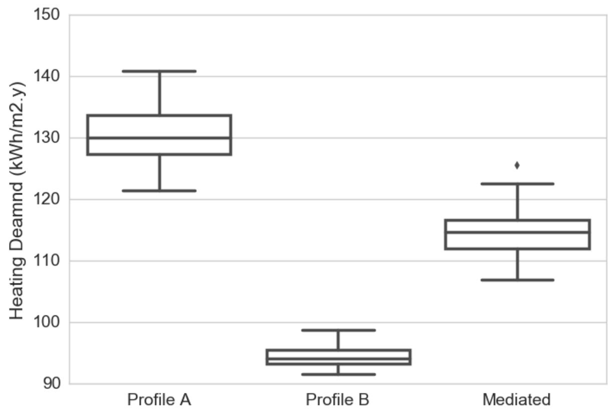


Figure 4: Heating demand for extreme window opening profiles simulations and the mediated simulations (100 replicates)

Stochastic Variations

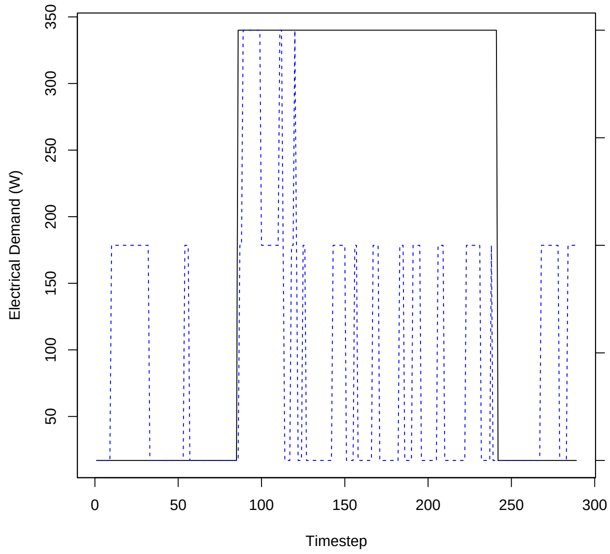
Depending upon the assumptions made in the choice of deterministic rules and schedules, performance results can deviate significantly from those arising from the stochastic representation of people. The predicted median heating demand for our office located in Nottingham obtained using No-MASS was $98.6kWh/m^2$, compared with $80kWh/m^2$ when assuming deterministic behaviours.
To this end our offices are assumed to be occupied according to the following fractional schedule during the weekdays: [0.0] 00:00 until 07:00, [0.25] until 08:00, [0.5] until 09:00, [1.0] until 12:00, [0.75] until 14:00, [1.0] until 17:00, [0.5] until 18:00, [0.25] until 19:00, [0.0] until 24:00. During the weekend offices are assumed to be vacant. Interactions with external shades and lights operate on the same schedule, with the windows being open when occupants are present and the indoor temperature exceeds 24° C.
Repeated stochastic simulations enable the likely range of possible energy demands arising from occupant interactions to be quantified. In the case of our office 90% of our predicted heating demand results Lie in the range $96kWh/m^2$ to $102kWh/m^2$ (see also Figure 3, in which we also plot the deterministic value as a point for comparison).

Belief-Desire-Intentions

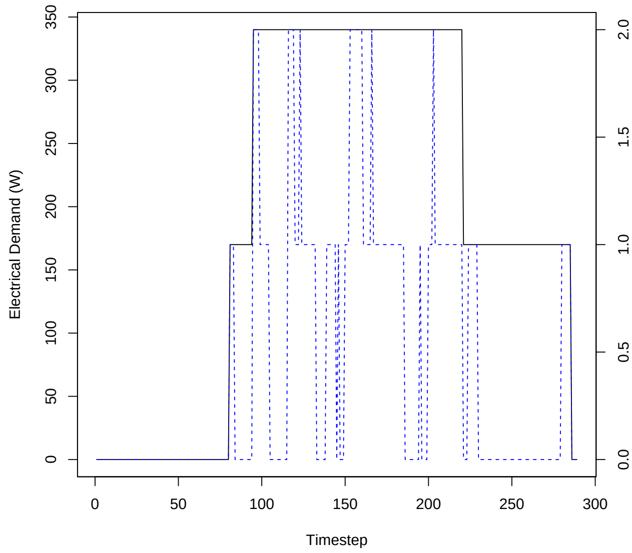
To determine whether our BDI rules are useful, we examine the impact on building performance of including a simple plan relating to the presence-dependent use of appliance in the workplace (as described earlier). This plan is executed in No-MASS each time we can achieve the goal of turning on the appliance at first arrival and turning off the appliance at the last departure.
The overall demands (heating and cooling) are compared for 100 replicates against a base case (where the appliance use are based on a deterministic occupancy schedule) using a t-test ($P < 0.005$). The BDI rules have a significant effect on both the heating and cooling demands of the office, when the rule is applied there is an increase of $2kWh/m^2$ in heating demand due to the lower usage of desktop computers, and a decrease of $2kWh/m^2$ in cooling demands. Figure 5 demonstrates the effectiveness of the BDI rules, in this instance without the BDI rule and using the appliance profile given in DesignBuilder, where from 7am till 8pm the computer is on; there are times when the stochastic agent is present but not consuming electricity. Although this may be the case, the BDI rule mimics the stochastic presence profile in a more realistic fashion. It is unlikely that a computer will use this constant demand throughout the day. With BDI rules it would be possible to build in a computer's energy saving profile, for example if an occupant is not present and 20 minutes has passed the computers power state could be reduced, effectively putting the computer into standby mode.

Social Interactions

To demonstrate our social interaction framework we assign coefficients to our agents windows opening models that represent two extremes in behaviour: Profile A representing highly active window opening usage and Profile B highly inactive usage. From Figure 4 we see that our relatively inactive agent with Profile B ($94kWh/m^2$ when occupying an office alone) negotiates a reduced window opening duration and therefore a lower heating demand than that of Profile A ($130kWh/m^2$ when occupying an office alone); so that the new negotiated median heating energy demand ($114kWh/m^2$) lies between the two extremes.

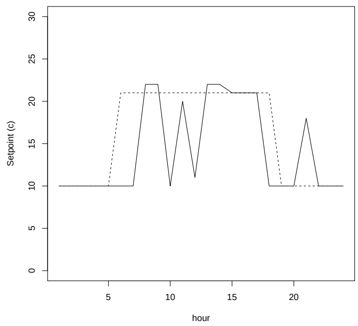


(a) Deterministic Profile

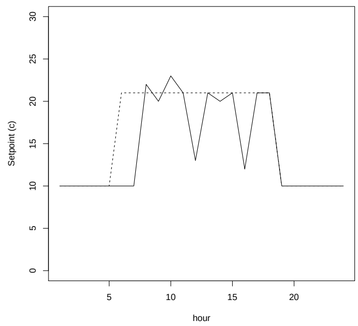


(b) BDI rule

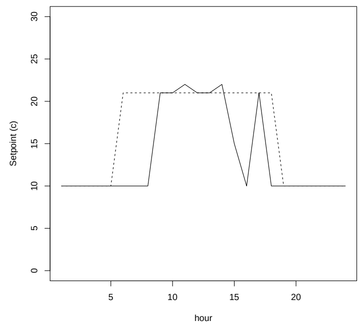
Figure 5: Stochastic occupancy profile (dashed line) with electrical demand from computer(s) (solid line)



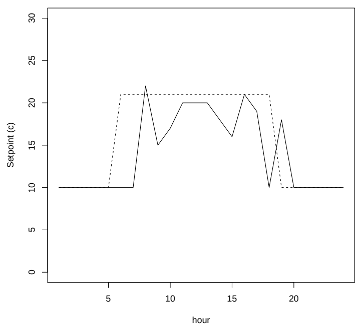
(a) Janurary



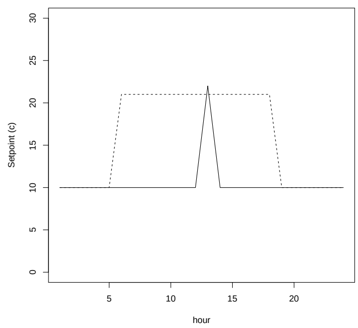
(b) Feburary



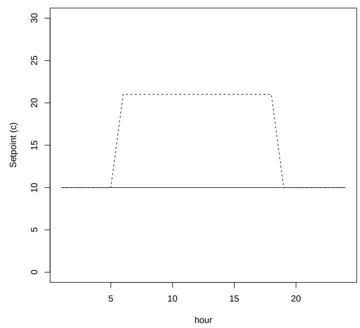
(c) March



(d) April



(e) May



(f) June

Figure 6: Learnt monthly heating setpoint profiles from 100 replicates, 6 months

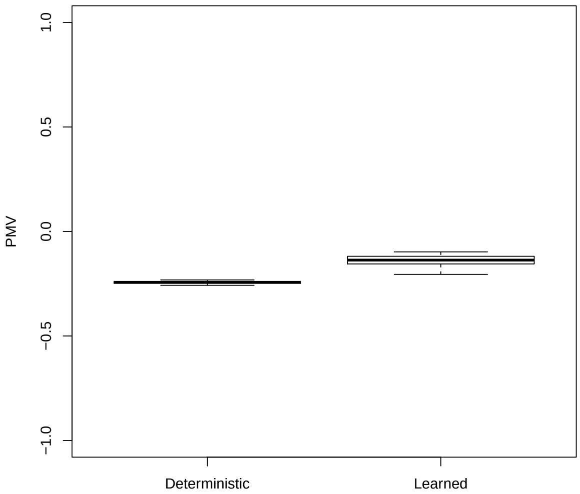


Figure 7: Box plots of 100 replicates for learning heating setpointsYearly Mean PMV

Reinforcement Learning

After a period of 19 training years a No-MASS agent learns the heating demand profiles presented in Figure 6. For the cooler months, November to March, similar profiles are learnt. At the end of this heating season in April and October the agents have learnt three distinctive profiles: heating first thing in the morning, reducing heating, increasing around lunch, reducing after lunch and increasing again before departure. During the summer months and September there is no desire by the agents to enable heating.

The deterministic setpoint schedules simulated a median heating demand some $5kWh/m^2$ larger than the learnt schedules. It might be expected that agents using the learnt schedules would experience more thermal discomfort, as they are using less energy to heat the building. However in Figure 7 we observe that this is not the case. Using the learnt profiles we have a yearly median PMV that is closer to zero than that achieved using the deterministic setpoint schedules. The learnt profiles effectively meet the agents energy and comfort aspirations.

Conclusion

In this paper we have described a new Multi-Agent Stochastic Simulation platform (No-MASS) that employs three complementary strategies to comprehensively model agents behaviours: (i) data-driven stochastic models where data is abundant, (ii) a belief-desire-intent framework to model data-scarce but simple interactions, and (iii) Agent-Learning for more complex data-scarce interactions. We also introduce a preliminary framework to emulate social interactions amongst the members of diverse populations. We have demonstrated the utility of these strategies using a simple hypothetical office building. Although comprehensive, there is nevertheless significant scope to extend this work to further reduce the performance gap between simulated and real world buildings.

Our current priorities relate to: (i) the probabilistic generation of households whose members socio-demographic characteristics relate to the characteristics of the house being modelling, (ii) the integration of models of large and small electrical appliances in homes and also (data-permitting) in workplaces, (iii) the complement of the model predicting short-term presence with one that predicts long-term absences due to illness, vacations and business trips. We are also keen to test (and improve upon) the validity of our simplified emulation of social interactions. We are similarly keen to establish the validity of the ensemble of models integrated into No-MASS; but conducting a high quality experiments to achieve would be challenging. A more pragmatic pathway would be to conduct a study similar to that of Blight and Coley (2013), where No-MASS is tested against buildings designed to the passivhaus standard. These buildings have stringent design and construction standards that must be kept to, so that any variations should be due primarily to occupants and / or weather conditions.

Finally, in this paper simulations with No-MASS have restricted to a single building. However No-MASS can in principle be generalised to handle multiple buildings to support integration with tools like CitySim (Robinson, 2011). Furthermore, the same multi agent stochastic simulation framework could be used in the simulation of smart grids, with appliances becoming agents, their demand profiles simulated with stochastic models where data is available and BDI/ agent learning when not. The appliance could communicate when to turn on using the social interaction model. The agent appliances would learn the optimal demand profiles for themselves for a day or week based on machine learning, either Q-learning or neural networks. A prototypical extension of No-MASS to support demand response strategies on this basis is described by Sancho-Tomás et al. (2017).

Acknowledgement

Jacob Chapman is supported by the Horizon Doctoral Training Centre at the University of Nottingham (RCUK Grant No. EP/G037574/1) and by the RCUKs Horizon Digital Economy Research Institute (RCUK Grant No. EP/G065802/1)

References

Axtell, R. (2000). Why agents?: on the varied motivations for agent computing in the social sciences. *Center on Social and Economic Dynamics Washington, DC*.

Baker, N. and K. Steemers (2003). *Energy and environment in architecture: a technical design guide.* Taylor & Francis.

Blight, T. S. and D. A. Coley (2013). Sensitivity analysis of the effect of occupant behaviour on the energy consumption of passive house dwellings. *Energy and Buildings 66*, 183–192.

Bonabeau, E. (2002). Agent-based modeling: methods and techniques for simulating human systems. In *Proceedings of the National Academy of Sciences of the United States of America*, Volume 99, pp. 7280–7.

Bordass, B., R. Cohen, M. Standeven, and A. Leaman (2001). Assessing building performance in use 3: energy performance of the Probe buildings. *Building Research & Information 29*(2), 114–128.

Ephrati, E. and J. S. Rosenschein (1991). The Clarke Tax as a Consensus Mechanism Among Automated Agents. *Proceedings of the National Conference on Artificial Intelligence*, 173–178.

Haldi, F. and D. Robinson (2009). Interactions with window openings by office occupants. *Building and Environment 44*(12), 2378–2395.

Haldi, F. and D. Robinson (2010). Adaptive actions on shading devices in response to local visual stimuli. *Journal of Building Performance Simulation 3*(2), 135–153.

ISO (2005). *ISO 7730: Ergonomics of the thermal environment Analytical determination and interpretation of thermal comfort using calculation of the PMV and PPD indices and local thermal comfort criteria*, Volume 3.

Jaboob, S. (2015). *Stochastic Modelling of Occupants' Activities and Related Behaviours.* Ph. D. thesis, The University of Nottingham.

Kinny, D. and M. Georgeff (1996). Modelling and Design of Multi-Agent Systems. *Proceedings of International Workshop on Agent Theories, Architectures and Languages*.

Kinny, D., M. Georgeff, and A. Rao (1996). A Methodology and Modelling Technique for Systems of BDI Agents. In *Proceedings of European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, pp. 56–71. Springer Berlin Heidelberg.

Nouidui, T., M. Wetter, and W. Zuo (2013). Functional Mock-Up Unit Import in EnergyPlus For Co-Simulation. In *Proceedings of International Building Performance Simulation Association.*

Page, J., D. Robinson, N. Morel, and J.-L. Scartezzini (2008). A generalised stochastic model for the simulation of occupant presence. *Energy and Buildings 40*(2), 83–98.

Pigg, S., M. Eilers, and J. Reed (1996). Behavioral aspects of lighting and occupancy sensors in private offices: A case study of a university office building. In *Proceedings of ACEEE Summer Study*, pp. 161–170.

Reinhart, C. (2004). Lightswitch-2002: a model for manual and automated control of electric lighting and blinds. *Solar Energy 77*(1), 15–28.

Robinson, D. (2011). *Computer modelling for sustainable urban design: Physical principles, methods and applications.* Routledge.

Sancho-Tomás, A., J. Chapman, and D. Robinson (2017). Extending No-MASS: Multi-Agent Stochastic Simulation for Demand Response of residential appliances. In *Building Simulation 2017.*

Shoham, Y. and M. Tennenholtz (1992). On the synthesis of useful social laws for artificial agent societies (preliminary report). *Proceedings of AAAI92*, 276–281.

Sutton, R. S. and A. G. Barto (1998). *Reinforcement Learning: An introduction.* MIT Press.

Turner, C. and M. Frankel (2008). Energy performance of LEED for new construction buildings.

Watkins, C. J. C. H. and P. Dayan (1992). Q-learning. *Machine Learning 8*(3-4), 279–292.

Wooldridge, M. and N. R. Jennings (1995). Intelligent agents: Theory and practice. *The knowledge engineering review 10*(02), 115–152.

Zhang, T., P.-O. Siebers, and U. Aickelin (2011). Modelling electricity consumption in office buildings: An agent based approach. *Energy and Buildings 43*(10), 2882–2892.