Forecasting fire development

with

sensor-linked simulation

Koo, Sung-Han

The University of Edinburgh

2010

A thesis submitted in partial fulfilment of the requirements

for the degree of Doctor of Philosophy

Declaration

This thesis and the work described within have been completed solely by Koo, Sung-Han at the BRE Centre for Fire Safety Engineering, the University of Edinburgh, under the supervision of Dr. Stephen Welch and Prof. José Luis Torero. Where others have contributed or other sources are quoted, references are given.

August 2010

Koo, Sung-Han

Acknowledgements

"A flight ticket to Edinburgh … £673. Rent for my flat…£625. A PC for my research … £524. The living costs in Edinburgh… tens of thousands pounds (thanks mum and dad!). Joyful time in the fire group…priceless!"

Firstly, I would like to thank Dr. Stephen Welch for his guidance and encouragement during my research, sometimes like my mum, sometimes like my brother. Another great acknowledgement is to my academic dad, Professor José Luis Torero, and to my academic grandpa, Professor Dougal Drysdale. Also thanks to Dr. Jeremy Fraser-Mitchell in BRE, an academic uncle and an industrial supervisor, for sharing his profound knowledge about CRISP and for inviting me into the source coding world.

A special thanks to all my colleagues at the fire group for their friendship and for giving me wonderful time. Especially to those who shared room John Muir G6, chatting with me, and teaching me essential Spanish such as "Estoy hambriento."

I would also like to thank the BRE Trust and the University of Edinburgh for the financial support during my research. The research in the thesis has formed a part of FireGrid which was co–funded by the Technology Strategy Board's Collaborative Research and Development programme. The work has made use of the resources provided by the Edinburgh Compute and Data Facility (ECDF) (http://www.ecdf.ed.ac.uk/). The ECDF is partially supported by the eDIKT initiative (http://www.edikt.org.uk/). Thanks to Stephen P., George, Gavin, Liangxiu for pleasant time during the FireGrid project, and to Peter Mundy in Xtralis for wonderful video footages taken from the FireGrid D7.4 demonstrator which always draw attention from an audience. All supports are appreciated.

I would like to thank my family, mum, dad, and my lovely sister, who has always supported me mentally and financially.

Finally to my wife, "마누라 고마워!".

Abstract

In fire, any information about the actual condition within the building could be essential for quick and safe response of both fire–fighters and occupants. In most cases, however, the emergency responders will rarely be aware of the actual conditions within a building and they will have to make critical decisions based on limited information. Recent buildings are equipped with numbers of sensors which may potentially contain useful information about the fire; however, most buildings do not have capability of exploiting these sensors to provide any useful information beyond the initial stage of warning about the possible existence of a fire.

A sensor–linked modelling tool for live prediction of uncontrolled compartment fires, K–CRISP, has therefore been developed. The modelling strategy is an extension of the Monte–Carlo fire model, CRISP, linking simulations to sensor inputs which controls evolution of the parametric space in which new scenarios are generated, thereby representing real–time “learning” about the fire. CRISP itself is based on a zone model representation of the fire, with linked capabilities for egress modelling and failure prediction for structural members, thus providing a major advantage over more detailed approaches in terms of flexibility and practicality, though with the conventional limitations of zone models. Large numbers of scenarios are required, but computational demands are mitigated to some extent by various procedures to limit the parameters which need to be varied. HPC (high performance computing) resources are exploited in “urgent computing” mode.

K–CRISP was demonstrated in conjunction with measurements obtained from two sets of full–scale fire experiments. In one case, model execution was performed live. The thesis further investigates the predictive capability of the model by running it in pseudo real–time. The approach adopted for steering is shown to be effective in directing the evolution of the

fire parameters, thereby driving the fire predictions towards the measurements. Moreover, the availability of probabilistic information in the output assists in providing potential end users with an indication of the likelihood of various hazard scenarios. The best forecasts are those for the immediate future, or for relatively simple fires, with progressively less confidence at longer lead times and in more complex scenarios. Given the uncertainties in real fire development the benefits of more detailed model representations may be marginal and the system developed thus far is considered to be an appropriate engineering approach to the problem, providing information of potential benefit in emergency response. Thus, the sensor–linked model proved to be capable of forecasting the fire development super–real–time and it was also able to predict critical events such as flashover and structural collapse. Finally, the prediction results are assessed and the limitations of the model were further discussed. This enabled careful assessment of how the model should be applied, what sensors are required, and how reliable the model can be, etc.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Computer simulation for emergency response

Safe and timely decisions for quick response of both fire–fighters and building occupants during fire incidents are one of the essential factors for minimising losses to life and property. However, emergency responders will rarely be aware of the actual conditions within a building and will often have to make critical decisions based on very limited information. Although fire–fighters may draw on their previous experiences of fires, the interpretation and extrapolation of conditions is still an extremely difficult task, even for the most experienced fire–fighters. A single misjudgement by emergency responders could lead to catastrophe; this was illustrated in the World Trade Center attacks, for example, when emergency crews continued operations totally oblivious to the impending collapse of the towers, and history is

littered with examples of tragic failures in emergency response resulting from simple lack of knowledge of this type [1].

Computer simulation has been widely used in fire safety engineering practice to represent the possible fire conditions in a building. Its applications also include risk assessment, smoke movement analysis, inferring the cause of the fire after the incident, improving life safety, sprinkler performance, and structural behaviour. Due to the complexity of fire, however, its capability to accurately reproduce fire phenomena has been a subject of debate for many years [2]. First of all, defining all the input parameters and boundary conditions for the simulation is extremely challenging. Even if we assume all the information required is somehow known, a very small change in the building or in the fire can affect the process and thus lead to the fire developing in a totally different way [3]. Secondly, a finite number of computer simulations cannot cover all the possible fire scenarios in a realistic building as there are typically numbers of enclosures where fire can potentially occur and even in a single room, there are many potential burning items and manners in which they might burn. Thus, in most cases, a small number of computer simulations, “selected cases”, are carried out, and only with a specific, “representative”, part of the building.

Studies on application of fire modelling tools to full–scale tests involving real fuels provide relevant insights [4]. Detailed simulations were performed both *a-priori* and *a-posteriori*, predominantly using Computational Fluid Dynamics (CFD). The results of the blind modelling produced *a-priori* by different teams of modellers of a fire scenario demonstrate that it is enormously challenging to predict the evolution of fire in realistic complex scenarios and that it is far beyond the current modelling capability [5]. But the results of *a-posteriori* CFD modelling further reinforce the difficulty of obtaining a good match to the conditions of the test even with the availability of models of the burning behaviours of the

individual items involved in the fire which had been reasonably well characterised in prior testing [6].

Modern built environments are rich in sensors which continuously collect data and monitor the building for various purposes such as fire detection, indoor comfort and security. In most buildings, however, this information is not usually provided to fire–fighters or building occupants beyond the initial stage of warning people that there may be a fire. Some sensors such as temperature sensors or smoke detectors can provide essential information about the fire and they could support emergency responders to have a better picture of what conditions they might face inside the building. More advanced buildings, e.g. “Intelligent Buildings” or “Smart Buildings” [7], which are equipped with more sophisticated sensors such as movement sensors or door sensors (open or closed), could even provide further details about the occupants, e.g. how many occupants are in the building and where they are located. These data can be exploited not only to support fire–fighters and occupants via useful guidance to safe activity and egress, but potentially also facilitating a novel way of exploiting fire models, by providing the information needed to run them in real–time. Temperature data, for example, can in principle be used to estimate the size and location of the fire and smoke flow data can help estimating the status of doors and windows (open or closed). All this information could potentially supply some of the “unknowns” in the fire model, thus assisting in maximizing the accuracy of the resulting predictions.

Studies on data assimilation, which functions by integrating between outputs of computer simulations and sensors, have been carried out in many engineering fields such as robotics, industrial control, and signal processing. In recent years, it has been exploited in other areas in order to build emergency response and management systems. Applications include containing chemical and nuclear pollutants [8], monitoring the risk of oil pollution [9], and tracking and visualising the predicted course of hurricanes [10]. Napolitano *et al*. [11] and

Mu *et al*. [12] developed a neural network model and a flow model to simulate the flood and the model is calibrated and adjusted during flood events. A real–time wave forecasts system [13] has also been developed which observes the wave at three different locations off the coast and feeds the information into a neural network to obtain forecasts of significant waves. There have been more sophisticated applications in weather forecasting, which rely on advanced statistical tools [14]. Furthermore these methods have been used in conjunction with computationally intensive field models which are based on CFD. In most cases, the studies show that the model performed "reasonably well" and the evolution of the outputs tends to agree with observed data [11][12][13].

Fire safety engineering practice, however, is still at the most basic stage of integrating sensors and computer models, in spite of various relevant efforts in recent years. Leblanc *et al*. used a zone model to predict the heat release rate time–history based on observations generated using the same model [15]. Notwithstanding the close reproduction of the heat release rate to the past history, the work does not really address how to produce a forecast of future events. Cowlard *et al*. [16][17] estimated the upward flame spread rate of Polymethyl–methacrylate (PMMA) based on visual recording of the flame height and temperature measurements. It is demonstrated that the input parameters in the analytical models could be optimised to the scenario in real–time, and that rapid convergence between the evolving experiment and the predictions was attained. Although the study has successfully shown the concept of super–real–time predictions steered by measurements, it is too simple for application to realistic fire scenarios where many other phenomena complicate the fire development. Davis *et al.* carried out a series of studies which present the whole process of developing a system integrating sensors in a building and computer simulation [18][19][20]. He first developed an algorithm which attempts to convert the sensor signals to the heat release rate (HRR) of the fire [18]. Once the HRR is known, a multi–room zone fire model, CFAST [21], is used to determine smoke layers and temperatures in other rooms in order to

provide the fire service an approximate overview of the fire scenario in the building [18]. The “sensor–driven” fire model then forms a part of a building system which controls the heating ventilating and air–conditioning (HVAC) system, the heating/cooling plant, and the security system [20]. These studies are interesting as they suggest a novel way of inferring HRR from sensors which are generally used in buildings; however, the current stage of the model does not provide any real capability for forecasting the future fire development.

One of the important distinctions between fire predictions and others such as weather forecasts is that the lead times for fire predictions are of the order of minutes while those for weather forecasts are of the order of days. Hence, application of sophisticated data assimilation schemes used in weather forecasting may not be viable for fire, unless truly massive computational resources can be found (and are affordable). It is also obvious that steering the fire model using sensor observations is not as simple as adjusting the steering wheel of a car. A single modelled fire scenario is generated by vast numbers of input parameters; however, most of the sensor data measured in the building cannot be directly used as input for the fire model. A sophisticated but quick way of optimizing the model and achieving a good match between the prediction and observations is required and there a host of practical difficulties in constructing this type of model. Moreover, finding optimum input parameters using sensor measurements is one of the challenging inverse problems which are typically ill posed, thus it may not have a unique solution.

In this thesis, a methodology is presented which could underpin a prototype emergency response system exploiting these real–time integrations between the fire model and sensors in the building. It is intended that the system will not only deliver information about the current fire but also provide estimates of the possible future development together with advance warning of extreme events such as flashover and structural failure. While “data assimilation” techniques provide formal procedures for reconciling predictions and

measurements, each with a degree of uncertainty, the emphasis here is more on *ad hoc* engineering approaches to the problem, reflecting the great variety in the potential sources of fire measurements and the highly complex nature of the development of real fires. The novel aspect of the modelling approach is adoption of a "steering" procedure using sensor measurements, such that the output of the model tends to become a progressively closer representation of the real fire, thus enabling some degree of "forecast" of evolving hazards.

The concept of combining sensing and modelling techniques was also proposed in a cross–disciplinary project, FireGrid [1] which was undertaken in parallel with this research. Relevant technologies such as advanced fire modelling, structural response and egress, sensor networks, grid infrastructure [22] and HPC were integrated in order to develop a prototype real–time emergency response system. The sensor–linked fire modelling approach presented in this thesis was embedded within the project, playing a key role as the 'simulation component' and requiring collaboration with other relevant disciplines such as delivering sensor data via the network and translating simulation output into operational instructions. Two sets of full–scale fire tests were carried out during the project, which gave opportunities to run the model 'live' and to obtain detailed information about the fire in a realistic building. The experimental data was also used in virtual playback mode after the test for model development and testing.

1.2 Thesis layout

The thesis consists of six chapters. The material is presented in the following form:

- Chapter 1 is a general introduction to the research.

– Chapter 2 introduces the FireGrid project and two full–scale fire experiments, the “Dalmarnock Fire Test" and the “FireGrid D7.4 demonstrator”. It describes the concept of the project and briefly presents the architecture and the components of a prototype emergency response system developed during the project. Detailed information about the two fire tests, are described including test building/enclosure descriptions and the instrumentations.

– Chapter 3 presents the generalised methodology to develop a sensor–linked model which can generate scenarios, adjust its input parameters, and provide some level of forecast of the fire development using sensor observations in real–time fashion. The chapter describes the choice of a fire model, a way of randomizing input parameters, matching criteria to evaluate the model outputs, and finally a sophisticated steering procedure based on a statistical method.

– Chapter 4 describes the results of the sensor–linked model from the application to two full–scale fire tests presented in chapter 2.

– Chapter 5 provides further analysis of the prediction results and discussion about the limitations of the methodology in terms of modelling strategy, matching criteria, optimising steering, and predictive capability.

– Finally, a summary of the work undertaken and outcomes of the study is presented in chapter 6 followed by suggestions for further works.

All the material in the thesis, i.e. the methodology to generate a sensor–linked model and the results of real–time fire prediction is created solely by the author. The full–scale fire experiment presented in chapter 2 was though carried out in collaboration with other participants in the project. The author was not directly involved in the “Dalmarnock Fire

Test” and the subsequent round–robin study. In the “FireGrid D7.4 demonstrator”, the author was involved from the early stage of the experiment preparation playing a key role when agreeing which sensors should be used, how the model should be run, and how the sensor data should be delivered to the model, etc.

Chapter 2

FireGrid and full-scale experiments

2.1 Introduction to FireGrid project

Limitations of dealing with emergencies have been exposed by tragic events such as Asian Tsunami disaster [23], Hurricane Katrina [24], and the terrorist attacks on the World Trade Center [25]. In most cases, post–disaster analyses raise the issue of whether the information that was available to the responders at the site was sufficient to tackle the emergency. Invariably the information is shown to be insufficient which leads to compounding failure and a potentially huge loss of life and resources.

The need for a new approach for tackling fires in buildings arose after a careful evaluation of the deficiencies in responding to fires emergencies in particular, considering cases such as

the World Trade Center [25], the tunnel fires at Mont Blanc and the Channel Tunnel [26], and numerous other recent incidents. The broad objective is to provide fire–fighters with as much useful information as possible that enables them to make sound and informed judgements while tackling the fire. The FireGrid concept was proposed by which it was envisioned that a novel infrastructure could be created to provide fire–fighters or occupants with critical and timely forecasts concerning the likely sequence of events following the outbreak of fire in a building. A desired consequence would be that the number of casualties and damage to property resulting from fires can be reduced [1]. The project focused on the development of a prototype emergency response system involving the following:

- Sensing: data from the emergency is collected and relayed
- Modelling: robust integrated simulation tools use this data to predict event evolution
- Forecast: simulations are achieved faster than real–time
- Feedback: processed results provided to active response systems/emergency services
- Response: an intelligent command and control system coordinates all intervention

These functionalities will be linked by the ‘grid’ technologies [22] that give the project its name, FireGrid. The grid technologies dynamically discover and co–ordinate distributed computing resources to work on solving a common problem [1].

The key benefits of the project will stem from the effective integration of the relevant technologies as shown in Figure 2.1, including: advanced modelling tools for fire, structural response and egress, sensor networks, grid infrastructure and High Performance Computing (HPC) resources to perform super–real–time computations, and decision support systems to translate simulation output into operational instructions for deployment of active systems and

intervention by human responders. The most challenging integrations include the ‘sensor computation integration’ and ‘simulation output filtering & Command/Control (C/C)’ and require a significant amount of further research. These issues are discussed at greater length in Welch et al. [1].

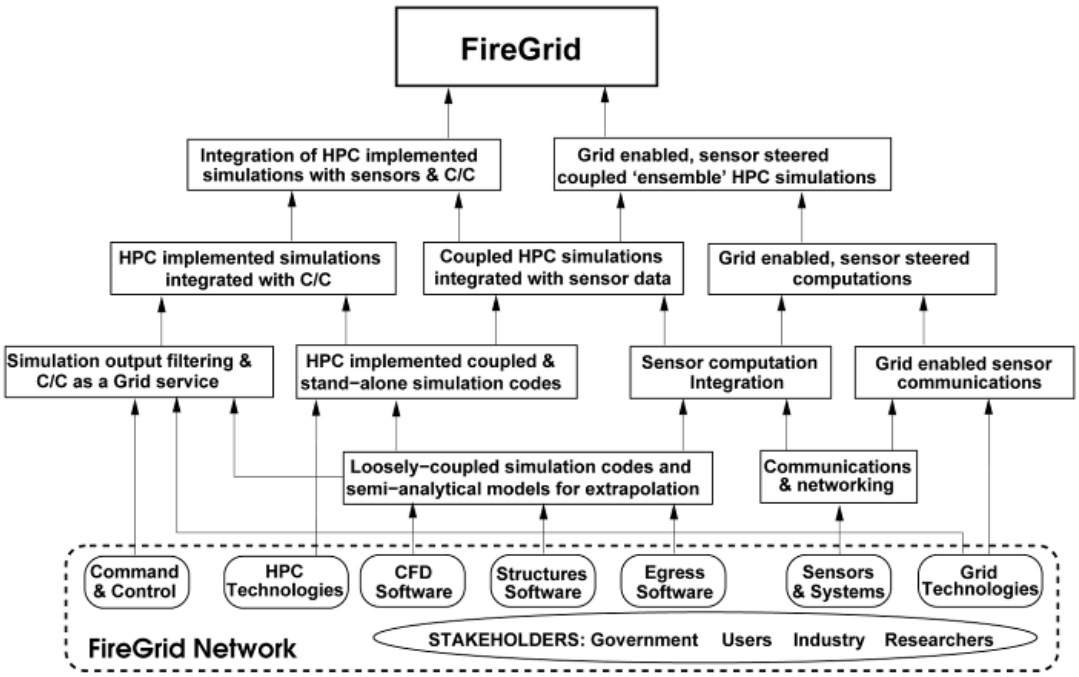


Figure 2.1 – Technology integrations required to develop FireGrid system [1]

The prototype FireGrid system consists of four principal components: a data acquisition and storage component for capturing and storing live sensor data; a simulation component for deploying and running computational models on HPC resources; a knowledge–based command–and–control component to provide decision–support for emergency responders; and a Grid middleware component to provide a uniform interface that connects the simulation component and the agent–based command–and–control component. Figure 2.2 shows simplified high–level system architecture and the interactions of these components [27].

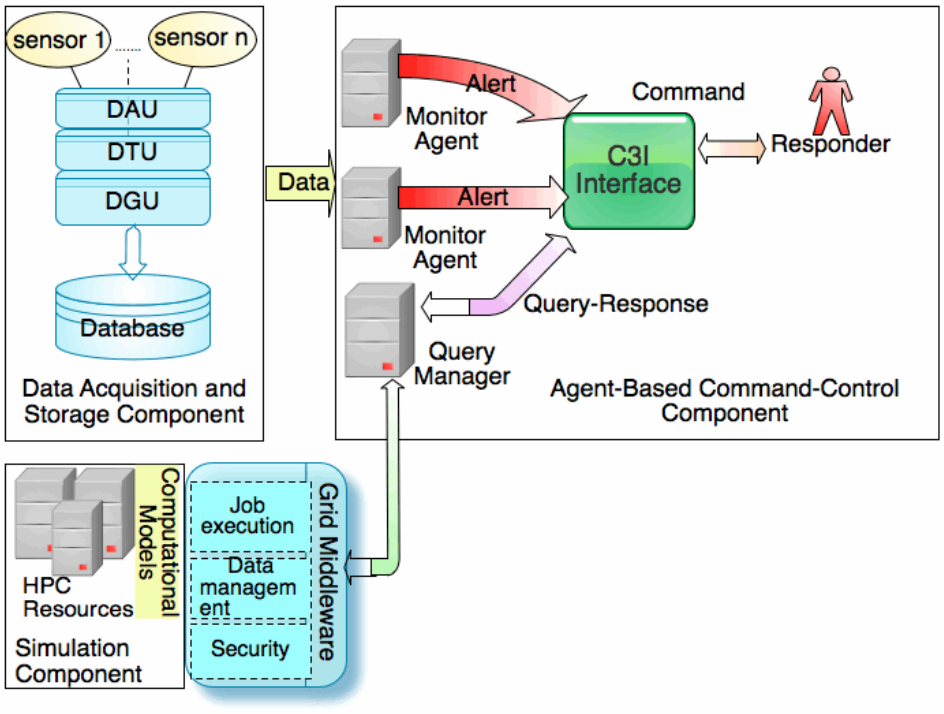


Figure 2.2 – Simplified high–level FireGrid system architecture [27]

The data acquisition and storage component is to implement the collection of data. The data will go through Data Acquisition Unit (DAU), Data Translation Unit (DTU), and Data Grading Unit (DGU) to capture and validate raw data from various types of sensors. Once DAU pulls raw sensor data from sensors, DTU will transform the raw data to a form that is appropriate for interpretation. Then DGU will filter the data attempting to validate its accuracy and reliability before storing it in the database [27].

The simulation component consists of computational models deployed on HPC resources for interpreting and predicting the current status and future development of fire and related phenomena. The model generates various outputs and it is steered towards the real fire using sensor measurements. HPC resources typically provide access to some number of processing units in parallel; hence code should be inherently parallel.

The ‘Agent–Based Command–Control Component’ consists of various ‘agents’ which carry out many key roles such as accessing the information generated by the models and formulating requests for specific information from users. It will also interpret and convert the ‘raw’ information to more intuitive format so that the emergency responders could easily understand the outputs and make appropriate decisions. There may be further agents in the system, used to monitor the status of environments such as a fire alarm agent that monitors sensor data for evidence of a fire and which can alert the user or perform some other actions accordingly.

From the fire safety engineering point of view, the key component in the FireGrid architecture is the ‘Simulation Component’, as this is where the fire model will be generating most detailed and “raw” information about the fire development. A sensor–steered fire model suggested in this thesis was used in the project as the ‘Simulation Component’. A simplified ‘Data Acquisition and Storage Component’ has been developed in order to offer pseudo live and pseudo real–time environment. By recording the sensor data and delivering it to the model in the same conditions as the live run, i.e. with the same time interval which had been used in the live test, the model can also be tested even after the real fire test.

Within the scope of the FireGrid project, two full–scale fire tests were carried out, the “Dalmarnock fire tests” and the “FireGrid D7.4 demonstrator”. The Dalmarnock fire tests consist of two parallel full–scale fires in a flat. These were deliberately set up in order to be representative of possible real fire events, with the fire flat being fitted with typical furnishings of a living room/office. During the preparation for the test, a round–robin study of blind predictions was also conducted in order to explore the *a-priori* predictive capabilities of fire modelling in realistic scenarios [5]. After the prototype FireGrid system has been established, a live fire demonstration, the “FireGrid D7.4 demonstrator,” that integrated all the components of the FireGrid project took place. A full–scale experimental

rig was built in a burn hall reproducing the internal layout of a small residential apartment. Although a prototype FireGrid system was built and performed successfully in the live test, it should be noted that the system is still under development. Further exploration and improvement is required as the performance of the system is inherently dependent on the prediction capability of the fire model being used.

Both tests and the round–robin study are presented in the following sections (2.2 and 2.3). The Dalmarnock fire tests results were used as default "real" fire data in the early stage of the sensor–linked model development and the simple scenario selecting procedure was tested using this data. A more robust model which includes an active 'steering' procedure was then proposed and it was tested live during the FireGrid D7.4 demonstrator. Further tests of the model were carried out in pseudo live mode using both fire tests results

2.2 Dalmarnock fire tests

Computer models have evolved considerably in the last decades, however, compatible large scale validation data has not been generated at the required pace. Most of the large scale tests were conducted in the 1970s and 80s and the objectives were usually to understand the different processes, including structural failures, or to validate simple engineering tools. In the early stage of the FireGrid project, this issue was raised and the BRE Centre for Fire Safety Engineering in the University of Edinburgh embarked on the mission of convincing the BBC to devote one of the episodes of their flagship documentary programme, Horizon, to the subject of Fire Safety Engineering. The Horizon show: "Skyscraper Fire Fighters" aired on BBC in the UK on April 24th, 2007 and the centrepiece of the documentary was the Dalmarnock fire tests.

2.2.1 Description of the test building and instrumentation

The fire tests were carried out in an apartment in a 23–storey residential tower in Dalmarnock, Glasgow in July 2006 [28]. The structure was built in 1964 comprising mostly cast in–situ reinforced concrete. There were six flats per storey, three either side of an access corridor which also led to two stairwells and elevator shafts. The two main experiments were held in identical apartments, located on different floors of the building but with the same layout and orientation, while the third, smaller test was held in one of the exit stairwells.

Figure 2.3 shows an isometric view of the apartment. The main structure geometry is shown in grey, window areas are represented in light grey and doorways in dark grey. Both flats used for the main tests were located on the north side of the building, facing westward. The apartment used in the first test was located on the 4th floor while the one used in the second test was located on the 2nd floor. The one storey gap between both tests prevented damaging the second apartment during the first experiment, protecting it from flames, smoke, heat and water seepage. The dwellings comprised a main flat corridor which led to two bedrooms, a bathroom and a living room which had an extra door leading to a small kitchen.

All existing furnishings and finishes contained in the apartments were removed, except for the embedded furniture in the kitchen and the doors. The kitchen door was removed with the original wall and was not replaced. Most of the fuel was concentrated towards the back of the compartment, away from the window, with a roughly even fuel loading throughout the rest of the compartment. This fuel configuration was intentionally very similar to the ISO room corner test (British Standards, 1993), allowing for entrainment during the fire to drive the flames against the flammable corner. The main fuel source in the experimental compartments was a two–seat polyurethane sofa (Figure 2.4), but they also contained two wooden office work desks with computers, each with its own foam–padded chair, three tall wooden bookcases, a short plastic cabinet, three small coffee tables and several paper and

tall plastic lamps. The bookcases were fully–laden with books, video tapes, paper–filled cardboard files and several other plastic items, as was the small cabinet. The fuel load density in the experimental compartment was estimated to be 32 kg/m^2 floor area, of wood equivalent.

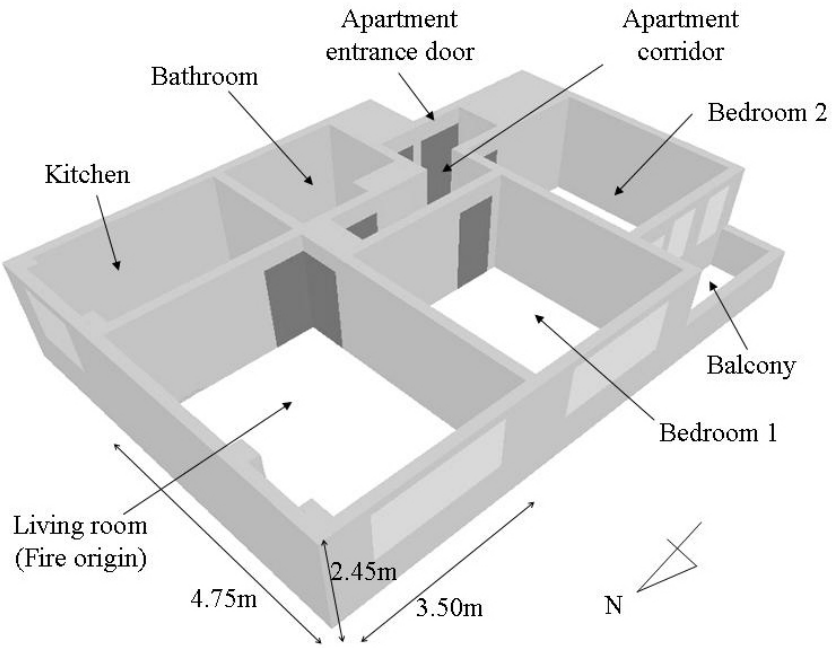


Figure 2.3 – North–west bird's eye view of the Dalmarnock fire test apartment layout [28]

(a) NW corner (b) NE heavily fuel-loaded corner

Figure 2.4 – Photographs of the Dalmarnock fire test compartment [28]

As shown in Figure 2.5, the experimental compartments were heavily instrumented, in order to obtain a high resolution of several different types of data associated to the development of both fires. The main types of instrumentation used throughout were thermocouples, heat flux gauges and laser smoke obscuration sensors, with bi–directional gas velocity probes installed.

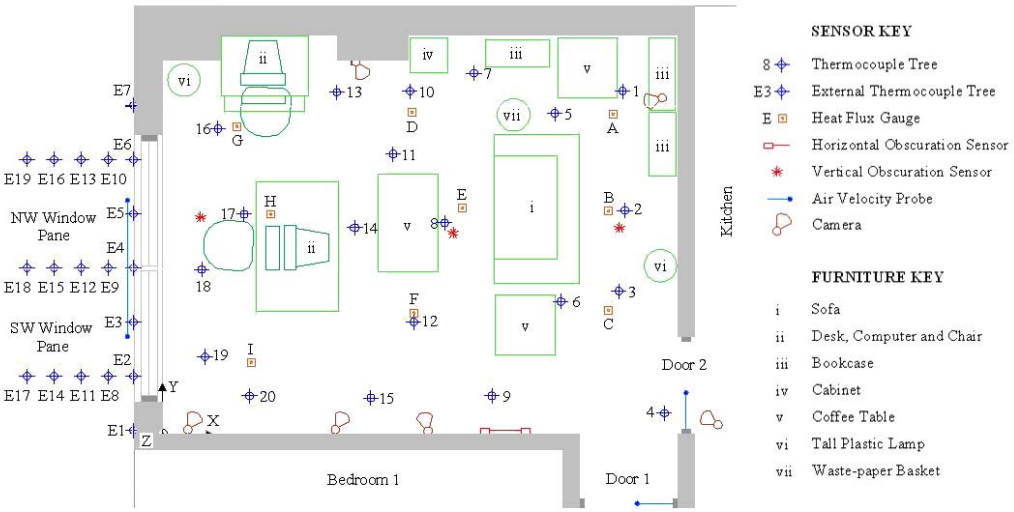


Figure 2.5 – Plan view and instrumentation of the Dalmarnock fire test compartment [28]

A series of thermocouple trees were used to measure the gas temperature. The thermocouples were places at distances of 0, 50, 100, 200, 300, 400, 600, 800, 1000, 1300, 1600 and 2000 mm from the compartment ceiling. The thermocouple trees were evenly distributed throughout the compartment such that they were not in contact with any solid objects, with the exception of the uppermost thermocouples in each tree which were in contact with the ceiling.

Additionally, a simple laser smoke obscuration sensor was used to measure light attenuation during the fire. Bi–directional velocity probes were used to characterise the gas flows in all openings to the main compartment (doors and windows). Nine heat flux gauges were placed on the ceiling of each experimental compartment and on the lightweight steel framing

kitchen partition wall. Network cameras were used as imaging devices to provide a visual description of the fire growth.

Two parallel fires were run in the Dalmarnock compartments (Test One and Test Two). While Test One was planned to allow a fire to develop freely to post–flashover conditions, Test Two was designed to allow for ventilation management. Although Test Two, the 'Controlled fire' successfully demonstrated the potential effectiveness of automated venting of smoke to hinder the build up of a smoke layer in a compartment fire, the results will not be further discussed as the test has no immediate relevance to the work done in this thesis.

2.2.2 Test One: The 'Uncontrolled' fire

Prior to ignition, data logging was commenced to record ambient conditions. 500 ml of heptane (accelerant) was poured into the waste–paper basket, soaking some of the crumpled up newspaper within it. A blow torch was used for pilot ignition of the contents of the waste–paper basket and the fire was allowed to grow unconstrained. A blanket draped over the sofa arm, dangling over the waste–paper basket, caught fire almost immediately. Four and a half minutes of sofa burning led to ignition of the contents of the bookcase adjacent to the sofa and waste–paper basket, close to the north–east corner of the room. Fire progressed up the bookcase followed by a flashover period which saw ceiling flame projection into the flat corridor, sudden reduced visibility in the corridor and simultaneous ignition of paper items in several locations. A summary of the major events of the fire is included in Table 2.1 below; the fire conditions are reported in detail elsewhere [29].

Table 2.1 – List of major events observed in the Dalmarnock fire test

Major events observed	Time (s)	Ventilation case
Ignition of waste bin beside sofa	0	–
Cushions ignite	9	–
Bookcase ignites	275	–
Fire engulfs bookcase (“flashover”)	300	Case 1
Flames project to apartment corridor ceiling	315	
Ignition of paper lamp and paper on table remote from sofa	323	
Kitchen window breakage (by heat)	720	Case 2
Living room window half breakage (by human intervention)	801	Case 3
External flaming	1080	–
Living room window breakage	1111	–

For the period up to ~275 seconds after ignition, a rough estimate of the possible heat release rate in the Dalmarnock test was obtained from the results of burning an “identical” sofa and bookcase under a furniture calorimeter. After the onset of the ventilation–controlled phase of the fire at about 275 seconds, information obtained from the bi–directional velocity probes installed in the internal doorways and main external window was used to deduce a rough estimate of the heat release rate (HRR), see Figure 2.6. The calculation, based on the principles of oxygen depletion calorimetry, assumes that all of the oxygen in the inflow air is consumed (23% air, by mass). This heat release rate estimate will tend to be an upper bound, particularly in the early period when fire temperatures are lower. Later, any overestimation will be countered by occurrence of some external flaming, which cannot be accounted for in the calculated value. The uncertainties in this calculation are hard to quantify precisely, but since they are expected to be relatively large they need to be taken into account [29]. The

approximate ventilation–control limits for the respective opening areas are also plotted in Figure 2.6 for comparison (see Table 2.1 for the definition of each ventilation case); these became progressively larger as the glazing failed, with a generally good correspondence to the measurements considering the known uncertainties.

The compartment average gas–phase temperature–time curve provides an indication of the general fire behaviour over time. Figure 2.7 shows the average compartment gas–phase temperature together with maximum and minimum temperatures which indicates the degree of temperature heterogeneity throughout the compartment. The general behaviour of the fire seen in the figure corresponds to the observed sequence of major events noted from camera footage.

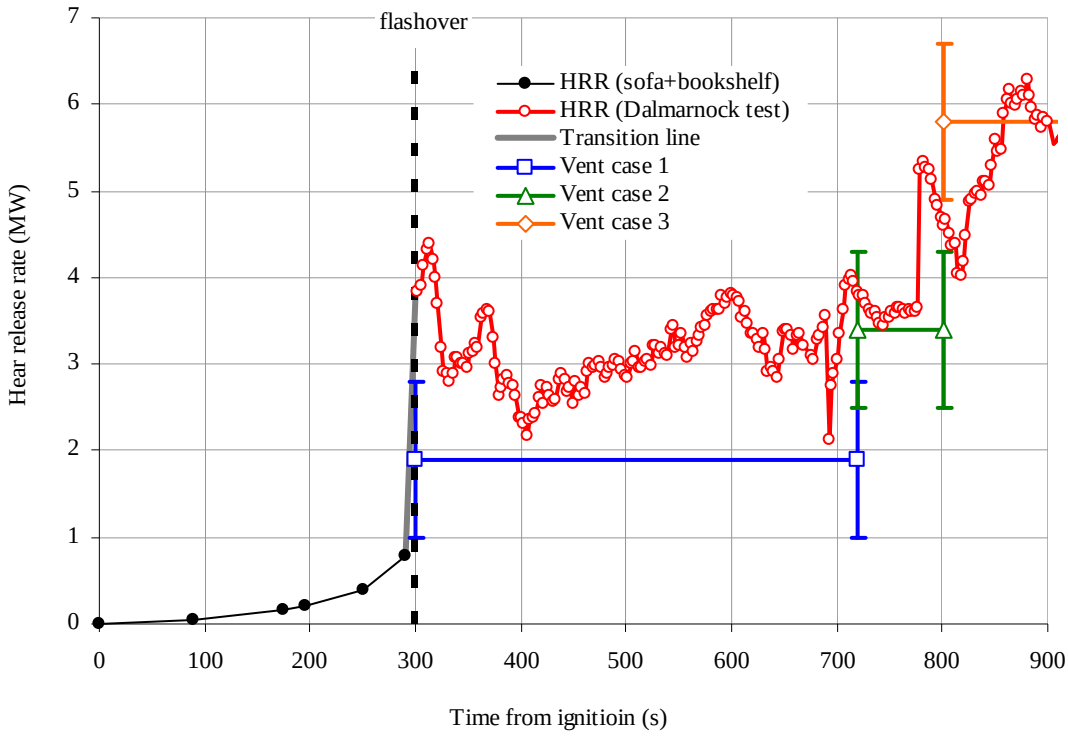


Figure 2.6 – Estimated heat release rate of the Dalmarnock fire test

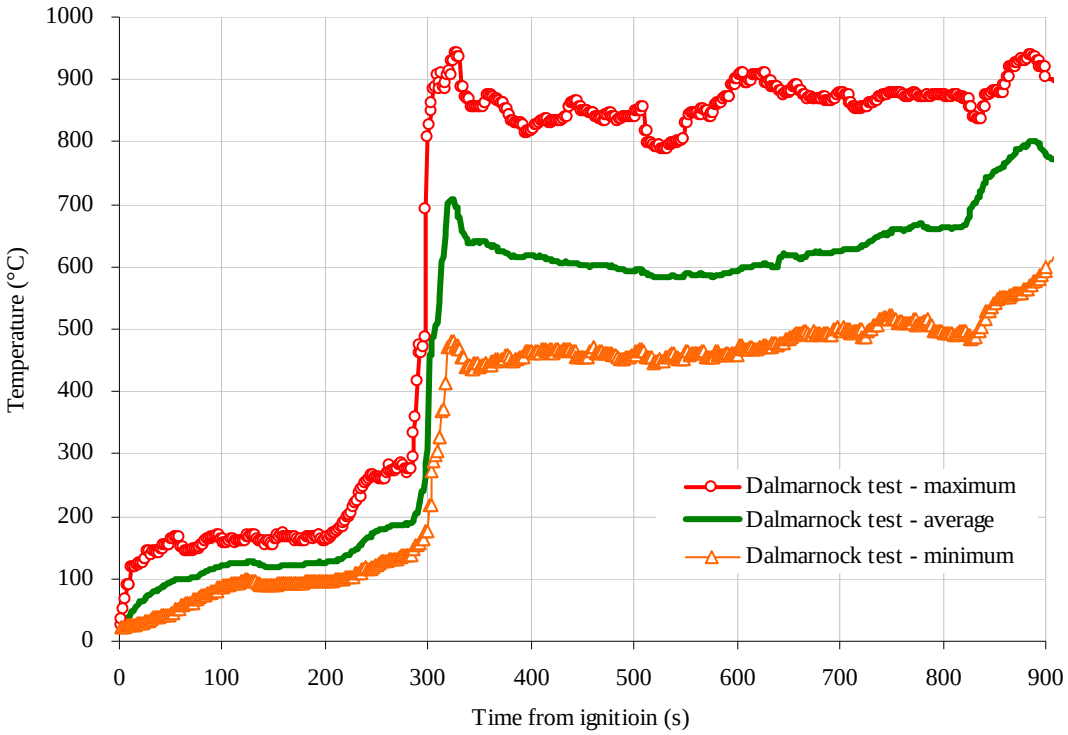


Figure 2.7 – Gas–phase average compartment temperature of the Dalmarnock fire test

The average temperature is seen to steadily grow for the first 100 seconds, reaching a small period of steady–state while the sofa is burning lasting about 100 seconds. As the fire begins to spread, average temperatures rise, leading to an overall slightly larger standard deviation as a larger proportion of the thermocouples are affected by the fire and smoke. A very steep rise in average temperature around 300 seconds into the fire is a clear indication of the flashover period observed soon after the fire spread to a fully–loaded bookcase in close proximity. This is followed by slight, gradual rise in average temperature which sees a further decrease in standard deviation just before the first compartment window pane was broken. This change in ventilation conditions leads to a second significantly steep rise indicating that the fire had become ventilation–controlled in the post–flashover period leading up to this event. Around 900 seconds into the fire, the average temperature reaches a small quasi–steady state period at peak average temperatures. Small variation in average temperature ensues, followed by fire brigade intervention at around 1140 seconds.

2.2.3 Round-robin study of *a-priori* modelling of the Dalmarnock Fire Test One

Before the Dalmarnock fire test were carried out, a round–robin study of blind predictions was conducted in order to explore the *a–priori* predictive capabilities of fire modelling in realistic scenarios [5]. The objective of this study was to compare the modelling results produced *a–priori* by different teams of modellers of the Dalmarnock Fire Test One. The participating teams were asked to attempt to forecast the test results as accurately as possible, and avoid engineering analysis with conservative assumptions of safety factors, as is common for use in fire safety design. The participants worked independently and had access to a large, common pool of data obtained from the initial conditions prior to the ignition of the fire. The information available to the teams included geometry and dimensions of the flat, room layout and information about furniture, photographs of the test compartment and the results of a laboratory experiment were the same sofa that acted as ignition source in the Dalmarnock tests had been burnt.

In total, ten simulations were submitted: eight CFD simulations using FDS4 [30], and two simulations using zone model, CFAST [21]. Each team was free to choose the model deemed most suitable or preferred for the task. The teams were asked to provide results in three ascending levels of complexity; i.e. ‘general fire behaviour and time to major events’, ‘transient fire behaviour by zones’, and ‘transient fire behaviour by fields, both in space and time’. The process of converting the data from CFD model–type to zone model–type information and the assumptions inherent to the process were the responsibility of each team and considered as part of the round–robin study.

As ‘general fire behaviour’, the predicted times to flashover varied approximately between a 180% over–prediction down to a 100% under–prediction and fell into two main groups, those predicted at 800±80 seconds and those that predicted flashover before 180±80 seconds. One simulation predicted that there would be no flashover. The predicted maximum average

temperatures in the compartment varied approximately between 50% over–prediction down to a 70% under–prediction.

Figure 2.8 shows the predictions and experimental estimate of global heat release rate (HRR). The simulations show a wide scatter of predicted fire behaviours, from 100% over–prediction to 30% under–prediction. The best average results and lowest scatter are obtained after the forced window breakage (at 800 seconds), as the teams were informed of the timing of this event.

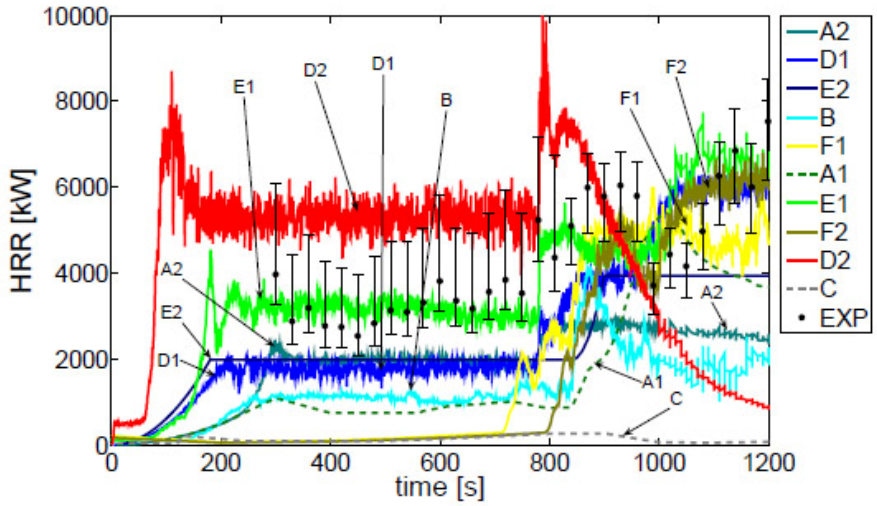
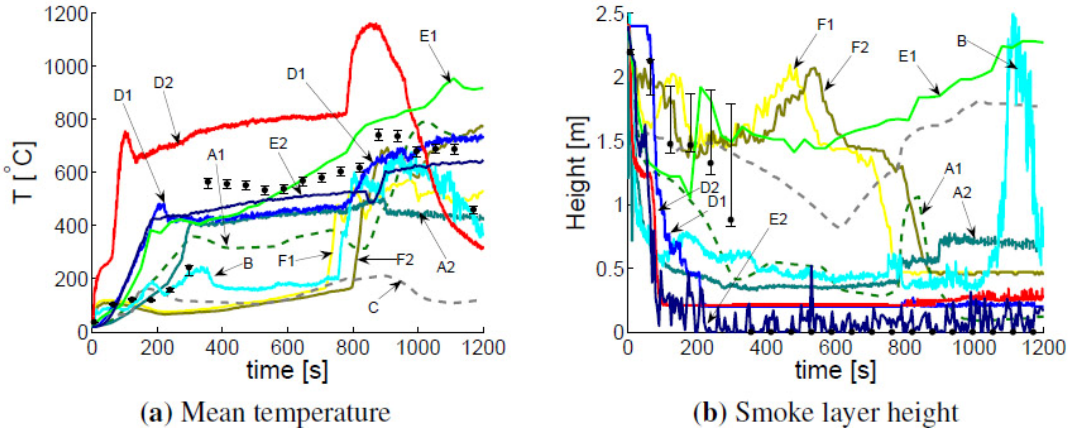


Figure 2.8 – Evolution of the global heat release rate within the Dalmarnock fire test compartment [5]



(a) Mean temperature **(b)** Smoke layer height

Figure 2.9 – Evolution of the hot layer in the compartment: (a) average temperature and (b) height to the interface from compartment floor [5]

Figure 2.9 shows the evolution of the average hot layer temperature (a) and the hot layer height (b) as predicted by the round–robin teams. The experimental values are averaged over the entire layer assuming that during the growth phase the interface is at the 100°C isotherm or at the height of the largest temperature gradient when below 100°C at the very beginning of the test. There is a wide scatter of modelling results shown in Figure 2.8. Most simulations under–predicted the hot layer temperature. Four simulations fell within a 10–40% under–prediction range and the others were above the 50% range of over– and under–predictions.

The results show large scatter and considerable disparity regarding the evolution of the hear release rate, both amongst the predicted fires and between the predicted fires and the experimental data. The scatter of the simulations is much larger than the estimated experimental error. The study emphasized the inherent difficulty of modelling fire dynamics and particularly fire growth in complex fire scenarios like the Dalmarnock fire test, and showed that the accuracy in (blindly) predicting fire growth (i.e. evolution of the heat release rate) is, in general, poor. However, since good predictions of the effects of a fire in the far field can be achieved, once the fire growth is known [2], it can be suggested, based on the results of this study, that some sort of forecast of fire development may nevertheless prove to be possible, if it takes into account feedback information from the evolving fire scenario.

2.3 FireGrid D7.4 Demonstrator

One of the deliverables (“Deliverable 7.4” gave the test its name “FireGrid D7.4 Demonstrator”) of FireGrid project was to setup and execute a large–scale fire test, in order to demonstrate the deployment of a complete instance of the prototype FireGrid system architecture. A full–scale fire experiment was carried out, together with some supplementary

tests. These were run in the state–of–the–art Burn Hall fire test facility at BRE Garston, near Watford, UK on October 2008.

2.3.1 Test rig description and instrumentation

The experimental rig for the demonstrator was intended to reproduce the internal layout of a small residential apartment of three rooms connected by a corridor, as illustrated in Figure 2.10. The rooms are referred to as follows: *Room 1* is the fire room, that is, it hosts the fire source(s); *Room 2* is the middle room accessed through the doorway halfway along the corridor; and *Room 3* is the furthest from the fire room at the opposite end of the corridor, connected to the outside through a second doorway. The final "room" is the *Corridor* itself. Each room – except the corridor – is a cube with inner dimensions of 2.4 m, and is connected to the *Corridor* through a door. The *Corridor* has inner dimensions of 3.6 × 1.2 m in plan and 2.4 m in height. All doors are of size 0.8 × 2.0 m.

The furniture layout in *Room 1*, the fire room, is also shown in the figure. The furnishings consisted of a two–seat sofa, which was the same as the one in the Dalmarnock fire tests, a television sitting on a coffee table and two bookshelves containing books, DVDs and magazines (similar to the main combustible items used in the Dalmarnock fire tests [4]). Two idealised structural components, two–dimensional assemblies consisting of a light–weight truss and a simple beam raised on supports, were located in the fire compartment in order to examine the fire–structure coupling behaviour and as an artificial representation of potential structural collapse issues.

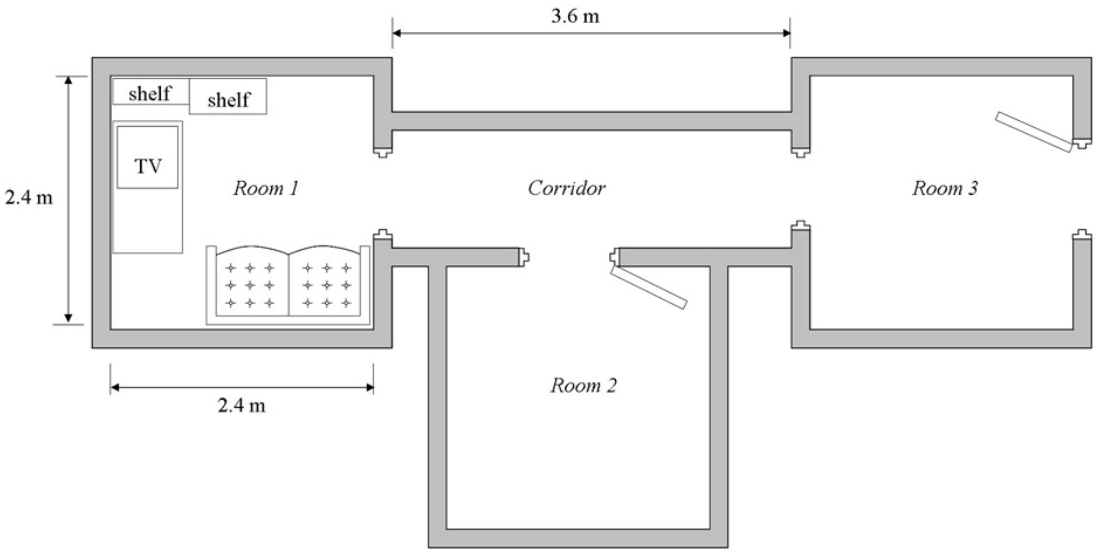


Figure 2.10 – Plan view of the FireGrid D7.4 demonstrator test rig and furniture layout

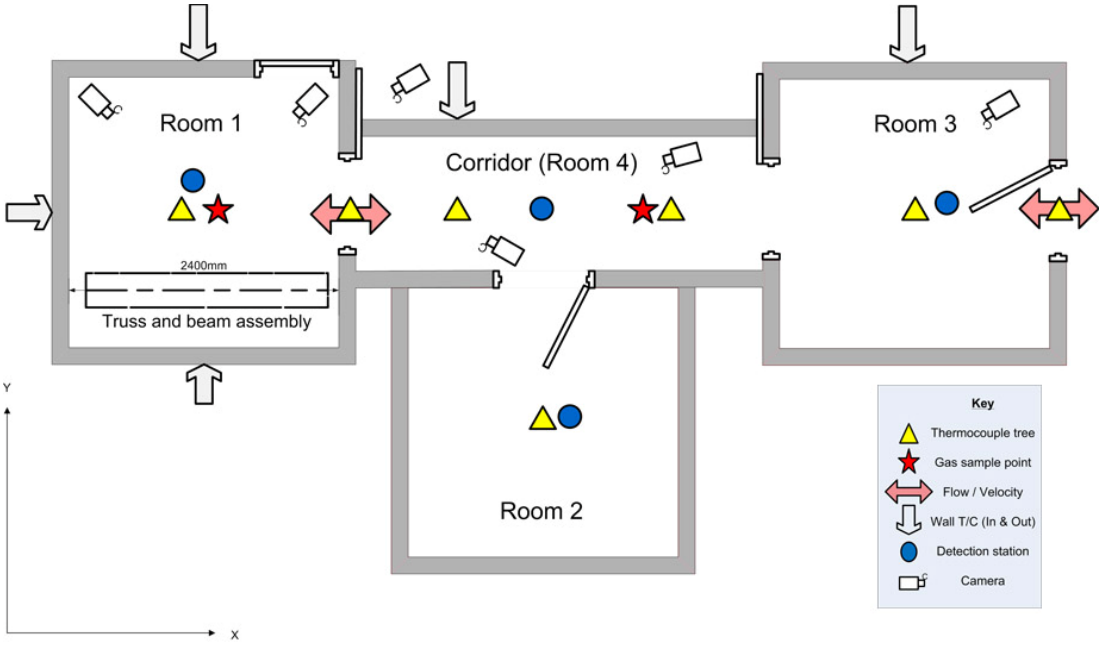


Figure 2.11 – Plan view and instrumentation of the FireGrid D7.4 demonstrator test rig [31]

Figure 2.11 shows the layout of the experimental instrumentation on the plan of the rig. A total of 125 channels of sensors were installed including gas and surface thermocouples, radiometers, gas velocity meters, and so on. Four centrally–located vertical trees of 11 thermocouples were installed in each room and six velocity probes were used in the doorways, three each at the opening connecting *Room 1* and the *Corridor* and the exit from

Room 3 to the external environment. Two radiometers were installed in *Room 1*. Gas sampling measurements were also made for O_2, CO_2 and CO concentrations at "nose height" (1.5 m) in *Room 1* and the *Corridor*.

Video surveillance of the experiment was comprehensive. Five cameras were monitored within the rig as indicated in the figure. An additional camera located just outside *Room 1* and a roving manually operated camera supplemented the cameras *in–situ* in the BRE Burn Hall to provide external monitoring of the rig.

2.3.2 Test results

Several smaller fires were run as part of the initial testing in order to demonstrate the early detection of fire using smoke detectors, to verify satisfactory operation of the FireGrid systems (that is: networking, data collection, sensors and the Grid) and to confirm that all instrumentation was working as expected. The final fire was intended to be a flashover, with a peak heat release rate in the range of 1–2 MW. For this test, furniture was present in *Room 1*, and all internal doors were open, as was the external door in *Room 3*. As per the Dalmarnock fire tests [29], the fire was initiated on the sofa seat and was expected to spread to other combustible items when the incident radiation heat flux on their exposed surfaces exceeded ~10kWm^{-2}, eventually leading to flashover in *Room 1* [31].

While visibility remained acceptable even four minutes after the first heat detector was triggered, the fire was then of a size that would be challenging to extinguish with a single portable extinguisher (see Figure 2.12). Ten minutes after the heat detector was triggered the smoke layer had descended to about 1.2 m and remained around this level for a further 10 minutes while the sofa was consumed (see Figure 2.13). Only when the fire transitioned from the sofa to adjacent items (the TV and shelves), at about 31 minutes after detection, did the fire begin to grow again; then within 2 minutes flames had reached the ceiling and 1

minute later the compartment was engulfed in flashover (see Figure 2.14). The fire was extinguished by manual intervention at about 47 minutes from detection and by the end of the experiment the fabric of the test rig was badly damaged and the loaded truss had collapsed (as it had been designed to when exposed to post–flashover temperatures).

Figure 2.12 – Four camera view of the FireGrid D7.4 demonstrator – 4 minutes from detection [31]

Figure 2.13 – Four camera view of the FireGrid D7.4 demonstrator – 10 minutes from detection [31]

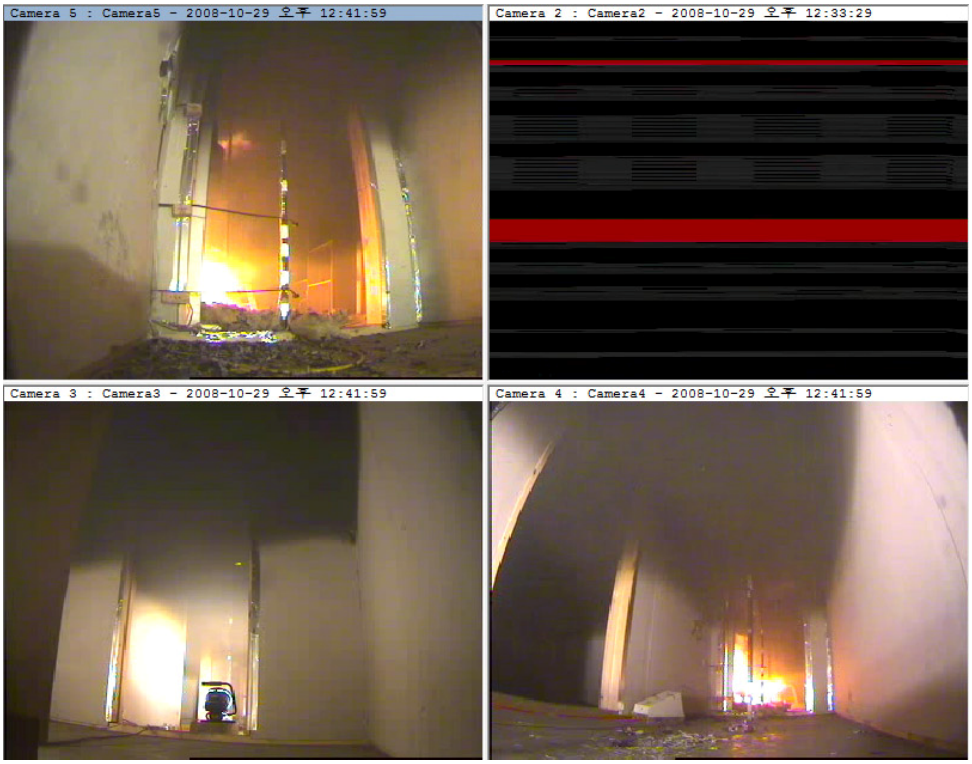


Figure 2.14 – Four camera view of the FireGrid D7.4 demonstrator – 33 minutes from detection [31]

The heat release rate measured by the calorimeter of the Burn Hall fire test facility and the gas–phase temperatures measured by the thermocouple tree in *Room 1* provide an indication of the general fire behaviour over time (see Figures 2.15 and 2.16). The temperatures rise up to 450°C until 1100 seconds from detection, then the fire starts to die down until about 1800 seconds. The temperatures level off for another hundred seconds and at around 2200 seconds from detection the fire starts to engulf the entire room. Although each thermocouple is spaced out 100 mm to 300 mm, thermocouple channels 1 to 9 show similar temperature values from the early stage of the fire while channel 10 and 11 appear to be in the cold layer until 700 seconds from detection. Then the temperature at thermocouple channel 10 rises rapidly but channel 11 keeps measuring relatively lower temperature throughout the fire incident until the flashover. As there is no significant temperature stratification, the average temperature of *Room 1* seems to represent the overall fire behaviour reasonably well.

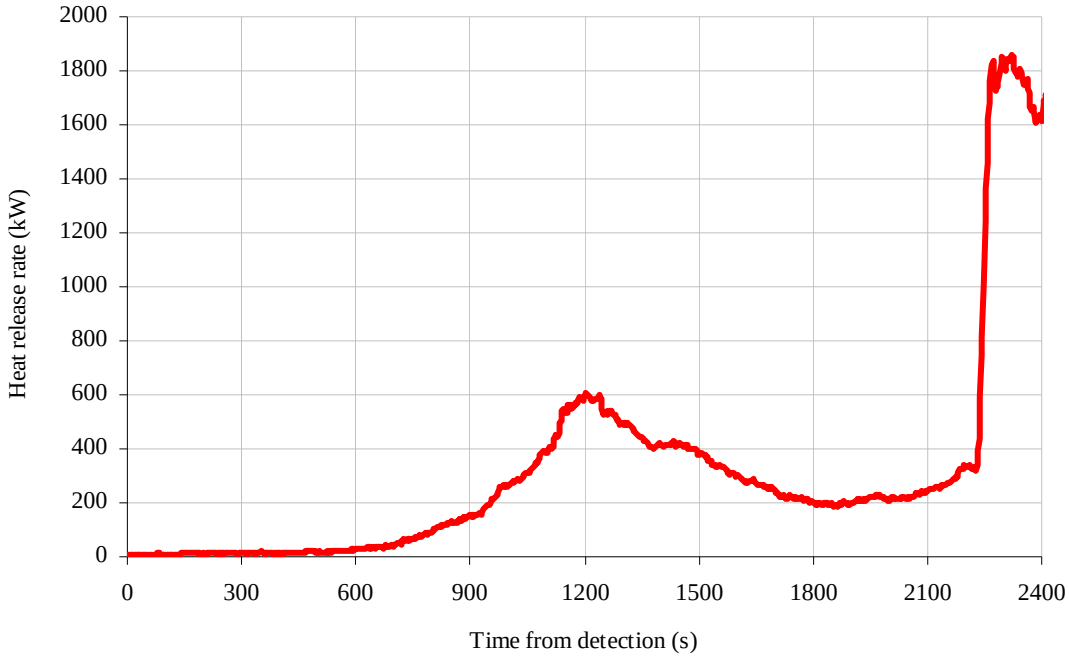


Figure 2.15 – Heat release rate of the FireGrid D7.4 demonstrator based on calorimeter data

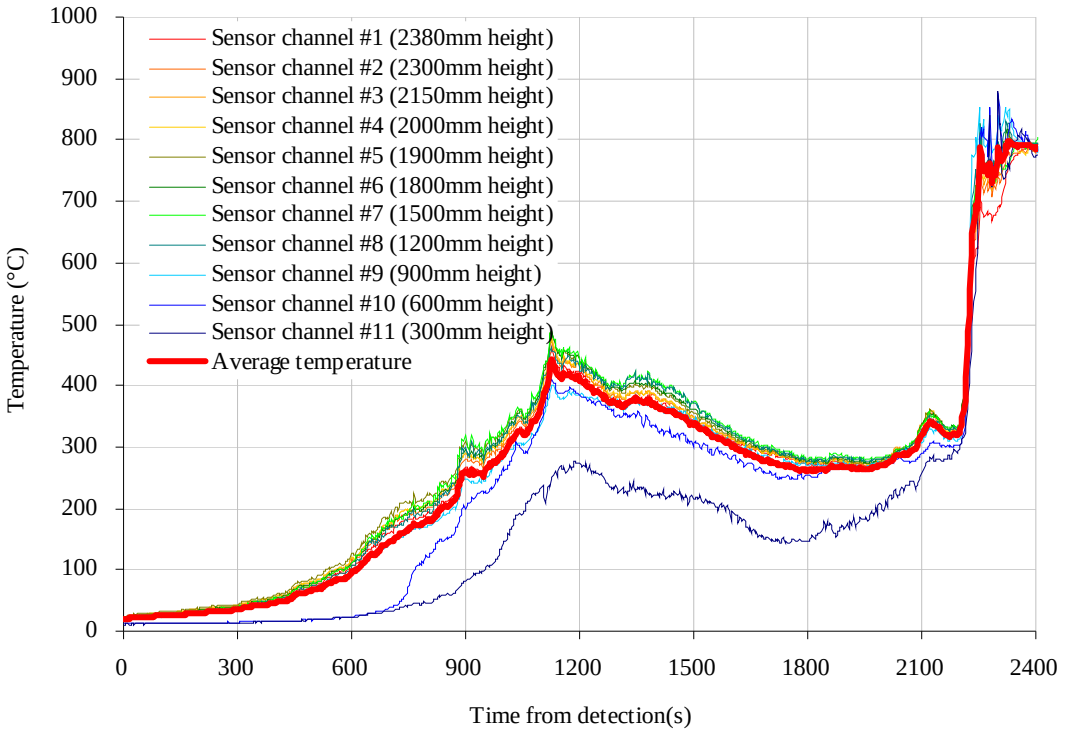


Figure 2.16 – Gas–phase temperature in *Room 1* of the FireGrid D7.4 demonstrator

Figures 2.17 to 2.19 show gas–phase temperature of *Room 2, Room 3,* and the *Corridor* respectively. Since *Room 2* does not have any windows or doors linked to outside, it was less affected by the hot layer coming out from *Room 1*, thus showing the lowest average temperature among all other rooms. Figures 2.17 and 2.19 show a similar temperature rise as *Room 1*; however, temperatures in *Room 3* are slightly lower than those in the *Corridor* as the smoke was cooled down by the cold air coming into the rig from outside through the door in *Room 3*. It is interesting that the cold air seems not only to cool down the temperature of the smoke but also to accelerate the stratification, as shown in Figures 2.18 and 2.19.

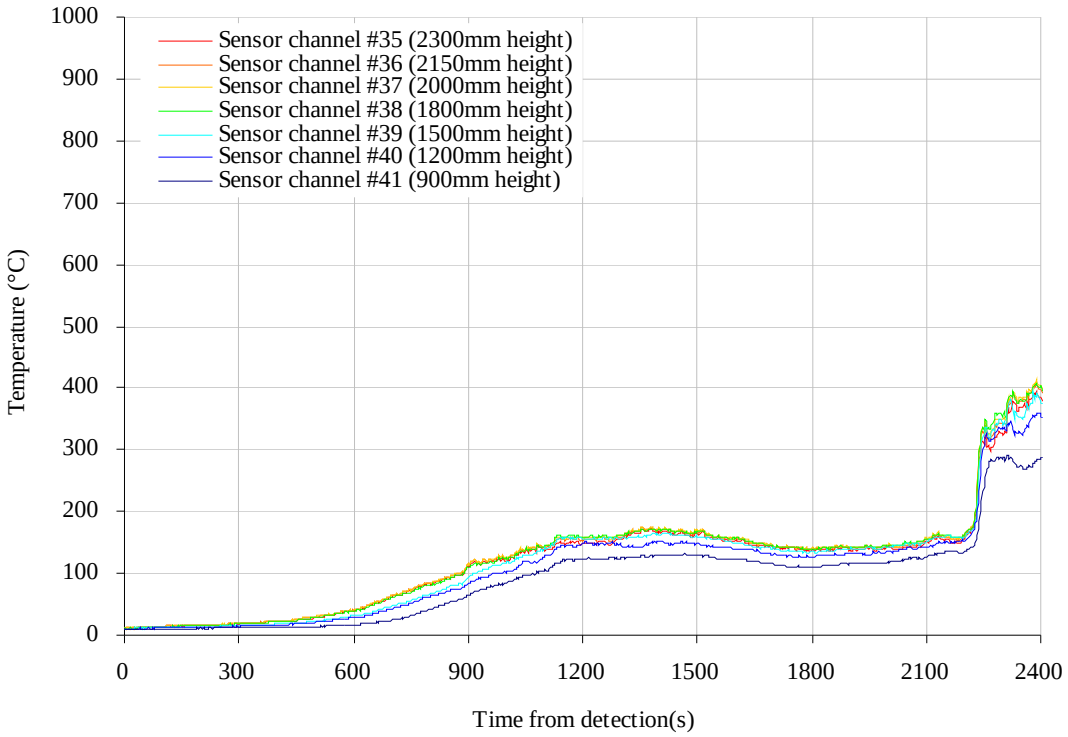


Figure 2.17 – Gas–phase temperature in *Room 2* of the FireGrid D7.4 demonstrator

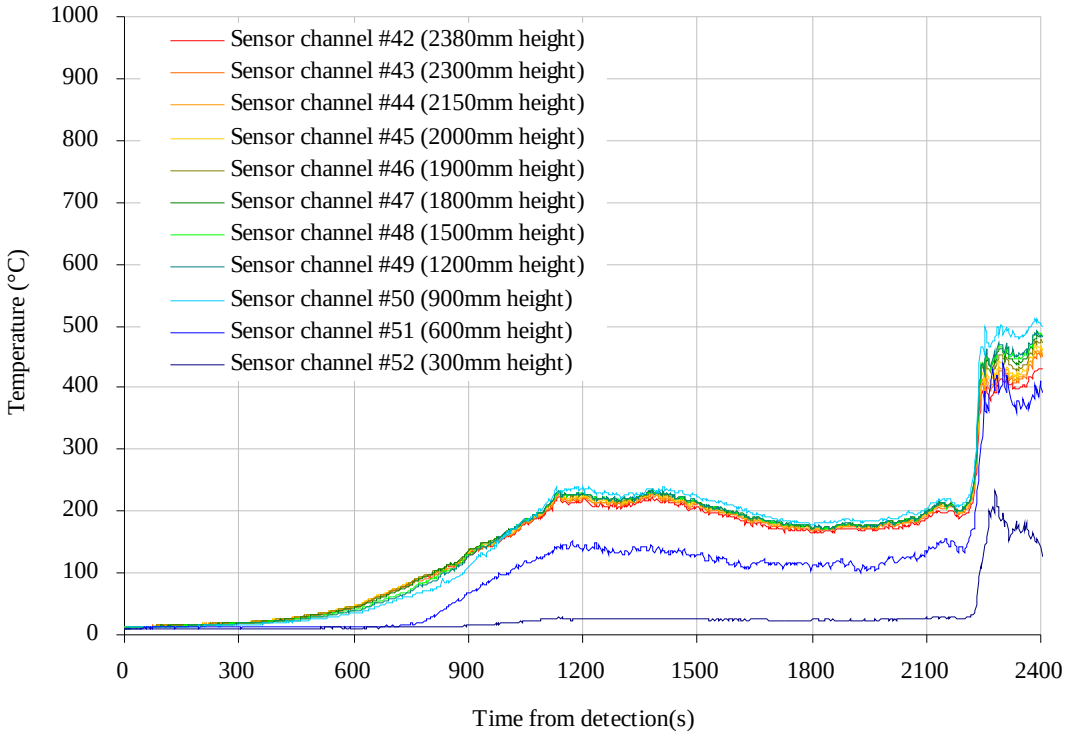


Figure 2.18 – Gas–phase temperature in *Room 3* of the FireGrid D7.4 demonstrator

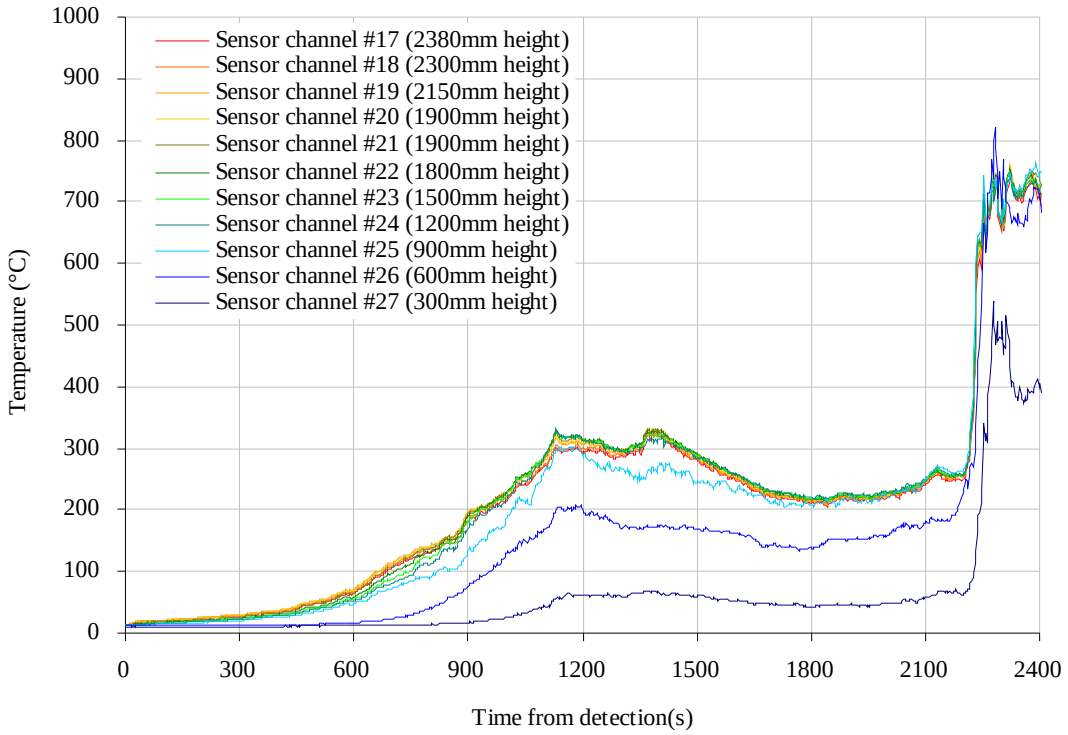


Figure 2.19 – Gas–phase temperature of *Corridor* of the FireGrid D7.4 demonstrator

As mentioned in the previous section, thermocouples were also installed at the doorways between *Room 1* and the *Corridor*, and between *Room 3* and outside. Most of the temperatures of airflows at the doorways were similar to the gas temperatures of the adjacent rooms, except sensor channel 12 as shown in Figure 2.20 (represented in red thick line). The sensor was not blocked or covered by any furniture during the experiment; therefore, it seems to be disconnected somewhere or to be in contact with a solid surface, thus measuring significantly lower temperatures than other neighbouring sensors throughout the fire. Nevertheless, as the temperatures were within the “acceptable” range, not giving some odd values such as below zero or greater than flame temperatures, the FireGrid system did not exclude this specific channel during the demonstration.

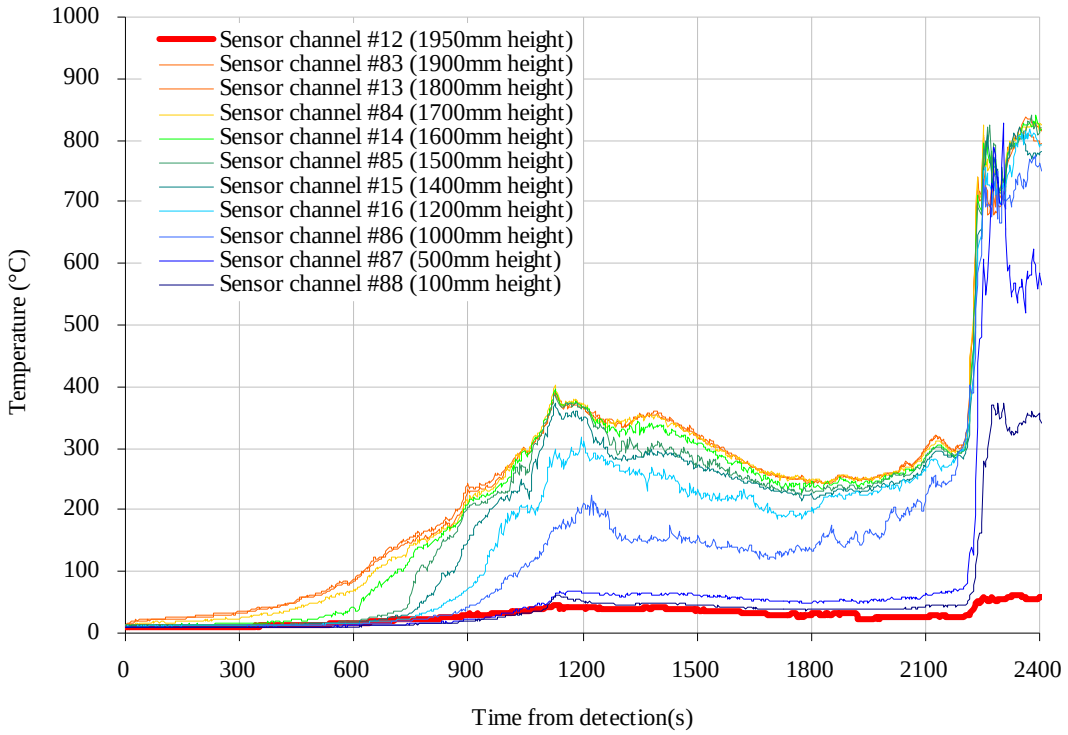


Figure 2.20 – Vent–flow temperature between *Room 1* and *Corridor*

The 'FireGrid D7.4 demonstrator' fire tests successfully developed into an unrestrained post–flashover fire. It has offered a great value in exhibiting very interesting fire behaviour, i.e. a fire which went to borderline of flashover following approximately a t^2 growth, then decayed with potential to die out, before the final flashover. The high density and broad range of sensors employed allowed for a complete characterisation of all aspects of the fire, including the pre– and post–flashover phases. All the measurements data obtained by 125 channels of sensors were stored in the database and can be used for virtual playback mode for the following research.

Chapter 3

Development of sensor-linked fire model

3.1 Introduction

In general, the process of defining input parameters for a given model consumes most of the effort and time being used for fire simulations. Parameters are usually defined based on any available information about the scenario, supplemented where necessary by the user's experience and knowledge on what might be reasonable approximations or assumptions. In order to use the model as an emergency response system, however, a different approach is required as the user will not generally have any information about the fire in advance. Even after the fire is detected, it is obvious that the information available in the early stage of the incident will not generally be sufficient for the fire model to reliably simulate the subsequent evolution of the scenario.

In this chapter therefore, a concept of randomizing input parameters for a model is suggested, using a simulation tool derived from a Monte–Carlo–based methodology. A randomised approach is required in order that the model will be capable of generating sufficiently diverse scenarios even without precisely defining their parameters. The model outputs will be continuously examined by comparing with sensor observations using statistical matching criteria. Different methods of establishing an appropriate feedback loop between the results assessment and the parametric space in which new fires are generated is suggested, with the aim of enabling some degree of "forecast" of evolving hazards.

CRISP, a zone model with capabilities for risk and egress modelling, was chosen as the basic simulation tool and the capability of the model in reproducing the gas layer temperature of the Dalmarnock fire test – given knowledge of the approximate heat release rate in the fire – was first verified. The randomised aspects of the model were examined and a matching criterion was selected to characterise the difference between the model predictions and the measurements. A sensor–linked modelling tool for live prediction of uncontrolled compartment fires, K–CRISP, was then developed in order to facilitate emergency response. The modelling strategy is an extension of CRISP, linking simulations to sensor inputs. The model was tested in 'real–time' using (pseudo) live sensor measurements obtained in the full–scale fire tests, the Dalmarnock Fire Test One and FireGrid D7.4 demonstrator. Next the CRISP model is described and then the methodology for integrating the fire model and sensor is suggested.

3.2 CRISP model

Choice of fire model is very important and the right balance between detail and practicality is one of the key factors to use computer simulation efficiently. In fire safety engineering

practice, various kinds of computer simulation models are being used as a fire can be analyzed in many different aspects based on the user's interest, such as smoke movement, egress, and structural behaviour.

The most advanced simulation techniques are based on Computational Fluid Dynamics (CFD), which provides the advantage of a great level of detail in the outputs but the drawbacks in terms of computational demands and complexity of implementation severely outweigh these and would render super–real–time predictions generally impractical for real fires. Moreover, fire predictions have relatively less lead time than other areas where CFD is widely used such as weather forecasting; thus sophisticated and demanding models, i.e. CFD, may not be appropriate for this type of fire application [32], at least for the foreseeable future.

Zone models, on the other hand, have many advantages in terms of flexibility and speed; therefore, it is clear that the efficiency of zone models renders them more applicable to adoption within an emergency response system for fire. The CRISP code [33] was selected as a basis for the study due to its various pre–existing capabilities for predicting fire and egress based on randomised inputs, as used in risk studies, though the new steering methodology would be amenable to use with other models adopting a similar approach.

The basic structure of CRISP is a two–layer zone model of smoke flow for multiple rooms, coupled with a detailed model of human behaviour and movement, thus providing a fully–coupled simulation of fire growth, smoke spread, active and passive fire protection systems, and the egress and interaction of people with the fire environment [33][34]. All the physical 'objects' are supervised by the Monte–Carlo controller, making each one perform for each time-step. The 'objects' include rooms, doors, windows, detectors and alarms, items of furniture etc, hot smoke layers, and people. The randomised aspects include starting

conditions such as various windows and doors open or closed, the number, type and location of people within the building, the location of the fire and type of burning item [35].

The burning behaviour is calculated as follows: firstly the flame height as a function of heat release rate and radius of the flame base is given by Heskestad [36]

$$h_F = 0.235\dot{Q}^{0.4} - 2.04R_F \tag{1}$$

and the configuration factor for a conical flame is given by Tien *et al.* [37].

$$F_{12} = \frac{1}{\sqrt{1+(h_F / R_F)^2}} \tag{2}$$

If the fraction of the fire's total heat output that is lost as radiation is λ, and assuming half of this fraction is radiated to the fire compartment and half to the burning surface, then the heat flux at the burning surface is

$$\dot{Q}'' = \frac{\lambda}{2} F_{12} \frac{1}{\pi R_F^2} \dot{Q} \tag{3}$$

The heat flux may be augmented by radiation from the hot gas layer in the fire compartment. The total heat flux then sets the target for pyrolysis rate per unit area (i.e. what would be achieved in a steady–state fire)

$$\dot{m}''_{tgt} = \dot{Q}'' / L_{vap} \tag{4}$$

where L_{vap} is the latent heat of vaporisation of the fuel.

The actual pyrolysis rate per unit area approaches the target by an exponential growth or decay as appropriate, with time constants τ_g and τ_d respectively, subject to the proviso that the target value is not overshot. When sufficient fuel has been consumed, so that the fire

enters the “burn out” phase, the target pyrolysis rate is set to zero. In this case, the heat release rate will tend to an exponential decay.

The pyrolysis rate for the whole fire is simply

$$\dot{m} = \dot{m}''_{act} \pi R_F^2 \tag{5}$$

The mass entrainment rate of air into the plume is then calculated, and hence the rate of oxygen entrainment (kg/s). If the stoichiometric ratio of the fuel is σ, the plume equivalence ratio is given by

$$\Phi = \frac{\dot{O}_2(plume)}{\sigma \dot{m}} \tag{6}$$

The yields (kg/kg fuel) of various combustion products are a function of the fuel type and the equivalence ratio; in particular the oxygen demand (kg/s) is

$$\dot{O}_2(demand) = \dot{m} Y_{O2}(\Phi) \tag{7}$$

and hence the value of the heat release rate for the next time step is

$$\dot{Q}(t + dt) = \Delta H_{O2} \dot{O}_2(demand) \tag{8}$$

where ΔH_{O2} is the energy release per kg of O_2 consumed.

It is assumed that the fire radius will increase at a uniform rate $\dot{R}_F$ (until the maximum radius is reached), which if the pyrolysis rate per unit area remained roughly constant, would tend to give an approximately t^2 fire growth curve.

By manipulating the equations above it is possible to estimate values for some of the parameters, given a heat release rate curve for the item burning with an unrestricted oxygen supply.

As it can be assumed that buildings will not normally be fitted with infrastructure to directly measure the heat release rate, i.e. the parameter which describes the behaviour of the fire most clearly, it may be necessary to make inferences about the fire development from any more accessible information. In this thesis, therefore, temperature was used as matching criterion assuming that continuous temperature measurements from the fire compartment are available.

3.3 Randomizing input data

The Monte–Carlo controller in the original CRISP code enables the model to use stochastically various input data from one run to the next. Functions are included to generate random numbers from any distribution. Although some input parameters for fire simulations are known to follow Log–Normal distributions [38], not all input parameters are well characterised. If the wrong distributions are used to define input parameters, the model may sample more (or less) values from specific parts of the distribution, thus resulting in more inefficient runs. Nonetheless, this is not expected to be a critical issue, because the use of the distribution is to enable the model to be sufficiently flexible and once the parametric space is narrowed to an optimum value, the influence of the type of distribution becomes negligible. The CRISP model can handle Normal distribution as well as Uniform or Log–Normal distributions, for simplicity it is assumed that the parameters follow only a Normal distribution for the study of this thesis In previous use of the code, randomized aspects were confined to the starting conditions, e.g. the status of various windows and doors (open or

closed); the number, type and location of people within the building; and the location of the fire and type of burning item. In order to vary the fire source definitions in particular, the user simply had to increase the number of burning item types and the model would select one item among them and ignite it. Therefore, further modification was made to the code to enable K–CRISP, the sensor–linked CRISP model, to read all of its input parameters in terms of probability density functions (PDFs), thus further generalising its applicability.

Parameters which were included in randomization were mainly those associated with the "fire model", such as burning item properties and full–room fire properties, together with others potentially influencing fire development, such as the thermal properties of the compartment boundaries. This provides a means of varying the fire source definitions in a more systematic manner than simply increasing the number of burning item types, i.e. varying each parameter individually during a simulation, though further generalisation could only be achieved by allowing ignition of multiple burning items. In addition, model capabilities were also extended in other relevant areas, for example, a structural integrity parameter was defined for specific building components based on pre–computed finite element analyses, invoked from a library of cases via a temperature look–up value. In turn the latter was derived either from a simplified thermal model driven by the upper layer temperature or provided directly by temperature measurements from the component.

Although most of the input parameters can be potentially randomized, some parameters such as building geometry could be predefined as they will be known prior to the incident. Note that material properties of the walls though are not fixed as it is usually not known and it affects the heat transfer, and therefore the smoke temperature, particularly when the fire has reached the fully–developed stage. As egress is not explicitly considered at this stage, all parameters related to human behaviour were also fixed. The status of doors and windows (open or closed) were predefined during both live tests according to the real condition;

however, they were fully randomised in the study of this thesis in order to assess the model's capability of inferring them from sensor observations.

Table 3.1 shows selected examples of item properties used for burning item. Most of the input parameters were set rather arbitrarily simply to have a sufficiently large parametric space, based on "guess" values of bounds within certain obvious physical constraints, e.g. the limits of the relevant geometry and possible material flammability properties, and it was intended that the model will try to adjust the distribution. The stoichiometric ratio, σ, was taken as 3.64 kg/kg (constant). There are many other factors which could also be stochastic, but for the purposes of this study were kept constant.

Table 3.1 – Item properties used in randomised input parameters

Parameter	PDF
Maximum radius, $R_{F,\max}$, over which fire may spread (m)	N(3, 2)
Height, h_{item} of burning surface (m)	N(0.5, 0.3)
Initial fuel load, $m_{item,0}$ (kg)	N(50, 30)
Fuel at onset of burnout, $m_{item,out}$ (kg)	N(10, 8)
Latent heat, L_{vap} of vaporization/pyrolysis for material (kJ/kg)	N(1390, 200)
Rate of flame spread, $\dot{R}_F$, across burning surface (m/s)	N(0.003, 0.002)
Exponential growth time constant, τ_g (s), when $\dot{m}''_{tgt} > \dot{m}''_{act}$	N(3.0, 0.5)
Exponential decay time constant, τ_d (s), when $\dot{m}''_{tgt} < \dot{m}''_{act}$	N(75, 10)

Finally, the model was modified to access the input data, i.e. PDFs, before each scenario is generated rather than using the same parameter sets throughout the run. This enables K–CRISP model to quickly update its PDFs and use them for future scenarios when any changes are made.

3.4 Evaluating model output

Assuming that the K–CRISP model has sufficient flexibility, and the probability density functions (PDFs) for the fire model parameters allow a sufficiently wide range of fire scenarios to be simulated, then Monte–Carlo sampling will (eventually) uncover combinations of model parameters that lead to a good match between calculated scenario consequences and sensor observations. Once an optimal parameter set has been generated for the current conditions, the more likely future scenarios would have input parameters that are closer to these values than other random variants. But with any realistic number of sensors, a perfect match has a negligible probability of arising "by chance", i.e. as a result of a fortuitous choice of the necessary input values. Indeed, how does one define a match for an analogue device such as thermocouple?

One way is to calculate the standard deviation, s, of the difference between observed and computed values:

$$s^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{n} \tag{9}$$

where O_i are the sensor observations for sensor $i = 1, 2, \ldots, n,$ and E_i are the corresponding K–CRISP predictions. The model will generate many scenarios at the same time as the fire is developing, therefore, the sensor data will only be available up to a certain point in the fire development. As more sensor data becomes available, the model will calculate the standard

deviation of all the scenarios generated up to that point and select the overall 'best' scenario(s) which has/have the smallest standard deviation value.

It is expected that standard deviation will identify the scenario giving a "good" match; however, it is desirable to have a more robust approach to differentiate between a match that is "reasonable" and one that is not, i.e. it is unsatisfactory to say that the best X% of runs done are "reasonable", since it is possible that even the best fit might be rather poor.

Another possible alternative might have been to calculate X^2 as in Equation 10 below,

$$X^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{\sigma_i^2} \tag{10}$$

where O_i are the sensor observations for sensor $i = 1, 2, \ldots, n$, and E_i are the corresponding K–CRISP predictions, then by definition [39] X^2 will follow a x^2 distribution with n degrees of freedom, provided that $O_i - E_i$ is Normally–distributed with variance σ_i^2 and each sensor reading is independent of the others. The goodness–of–fit over a time period (30 s) is computed since not all sensors may provide fresh readings at precisely the same time. The subscript i can therefore refer to different sensors or different points of time within the period of interest.

As the number of degrees of freedom, ν, (i.e. the number of sensor channels being used, multiplied by the number of time points for measurement) increases, the x^2 distribution can be approximated to a Normal distribution of mean ν, variance 2ν. This makes it very easy to compute the probability that the value of the x^2 distribution will be larger than X^2:

$$\alpha = 1 - \Phi\left(\frac{X^2 - \nu}{\sqrt{2\nu}}\right) \tag{11}$$

where $\Phi(z)$ is the Cumulative Density Function (CDF) of the Normal distribution. This probability α is the goodness–of–fit measure, and defines the weight assigned to each scenario in the steering procedure described in the next section.

A simple example may explain this better. Assume that there are 50 channels of sensors. K–CRISP has generated two different scenarios and the X^2 value of each scenario is 34 and 63, respectively. As mentioned above, the x^2 distribution can be approximated to a Normal distribution of mean 50 and the variance 100, as there are 50 degrees of freedom in this case. The probabilities of the x^2 value being bigger than 34 and 63, ($1-\Phi(1.3)$ and $1-\Phi(1.6)$) is approximately 0.95 and 0.10, respectively. Hence the scenario giving X^2=34 is a much better match to the observations than the scenario giving X^2=63.

3.5 Steering procedure

The K–CRISP model will generate various scenarios using randomized input parameters described in section 3.3. The matching criterion shown in section 3.4 will then be used in evaluating each scenario allowing identification of the combinations of input parameters that lead to a sufficiently good match between K–CRISP predictions and sensor observations. In order to feed this "knowledge" into the model enabling the model to "learn" it is proposed that a Bayesian inference procedure is adopted to modify the prior estimates for the input PDFs in the light of the most recent sensor data, as described below.

3.5.1 Bayesian inference

Bayes' theorem is most often expressed in the following form:

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)} \tag{12}$$

where *P(A)* is the probability of event "*A*" occurring (often termed the "prior" probability), *P(A|B)* is the conditional probability of *A* occurring, given that event "*B*" occurs (often termed the "posterior" probability). Similarly, *P(B)* is the probability of *B*, and *P(B|A)* is the conditional probability of *B* given that *A* occurs. The "prior" probability is updated in the light of fresh evidence (the occurrence of event B) to give the posterior probability.

If the events *A* and *B* both have range of different outcomes, then following the observation of particular outcome *B*, Bayes' theorem can be written as follows:

$$P(A_j \mid B_i) = \frac{P(B_i \mid A_j) \cdot P(A_j)}{\sum_k P(B_i \mid A_k) \cdot P(A_k)} \tag{13}$$

where the denominator is an alternative way of expressing *P(B)*. If the "likelihood function" *L(B|A)* is defined as

$$L(B_i \mid A_j) = \frac{P(B_i \mid A_j)}{\sum_k P(B_i \mid A_k) \cdot P(A_k)} \tag{14}$$

then Equation 13 can be simplified to

$$P(A_j \mid B_i) = L(B_i \mid A_j) \cdot P(A_j) \tag{15}$$

i.e. the posterior probability = the prior probability × the likelihood function. This calculation of the posterior probability estimate is known as "Bayesian inference".

For our purposes, the events B_i represent the sensor observations taking certain values b_i, and the events A_j represent the model input parameters taking certain values a_j.

A simple example may illustrate the process. Suppose there is a fire in a room, adjacent to a corridor that contains a smoke alarm. Further assumption can be made that the chance of detecting the fire at a given time is 50% if the connecting door is open, but only 1% if the door is closed (some leakage of smoke around the edges may occur). The notation can be used, i.e. A_1 = door open, A_2 = door closed, B_1 = detection, B_2 = no detection. If no prior information regarding the state of the door is given, it is assumed $P(A_1) = P(A_2) = 0.50$. Substituting into Equation 14, the likelihood $L(B_1|A_1)$ of detecting the fire if the door is open is 0.50 / (0.50 × 0.50 + 0.01 × 0.50), which equals 1.96. Hence, from Equation 15, the inferred probability that the door is open, given that the detector operated, is 1.96 × 0.50 = 0.98. If on the other hand there is no detection, the likelihood $L(B_2|A_1)$ of not detecting the fire if the door is open is 0.50 / (0.50 × 0.50 + 0.99 × 0.50), which equals 0.67, and hence the inferred posterior probability that the door is open is only 0.33.

3.5.2 Calculating the likelihood function

In a more realistic and complex building the link between cause and effect will be less straightforward. It cannot simply be stated, as above, what the likelihood function will be, but instead we need to estimate it. Let us suppose we have a situation where it is completely obvious whether or not a simulated scenario matches the observations (i.e. the goodness–of–fit, α, may only be 0 or 1). If we define $N(B_i|A_j)$ as the number of times the model predicts an observation B_i when the input parameter value is A_j, $N(B_i)$ as the number of times the model predicts an observation B_i regardless of the input value, $N(A_j)$ as the number of times the model simulates a scenario using the parameter value A_j, and N_{tot} as the total number of simulations, then the likelihood function as defined in Equation 14 can be calculated as

$$L(B_i \mid A_j) = \frac{N(B_i \mid A_j) / N(A_j)}{N(B_i) / N_{tot}} \tag{16}$$

Hence the calculation of the likelihood function requires no information other than the simulation model outputs. The model therefore needs to run (many times, very quickly) at the same time as the fire is developing, in order to calculate the likelihood functions for the observed sensor measurements.

Sensors such as thermocouples cannot be expected to match the model prediction in the same manner ("yes/no") as in the simple example above. However, rather than counting matching scenarios, we can sum the goodness–of–fit values instead in order to carry out an equivalent calculation

$$L(B \mid A_j) = \frac{\sum_{k=1}^{N(A_j)} \alpha_k (B \mid A_j) / N(A_j)}{\sum_{n=1}^{N_{tot}} \alpha_n (B) / N_{tot}} \tag{17}$$

where $\alpha_k (B \mid A_j)$ is the goodness–of–fit of the k^{th} scenario to use a parameter value of A_j, and $\alpha_k (B)$ is the goodness–of–fit of the n^{th} scenario (regardless of input parameter value).

The strategy chosen for model steering in the FireGrid project represents a pragmatic approach which was judged to provide an appropriate balance for the envisaged initial applications and available computational resources, though different implementations could also be used in principle. A choice is required on the type of PDFs to be used for various parameters, and how these PDFs are going to be steered. As mentioned in the earlier section, the K–CRISP code can handle various types of PDF, such as Uniform, Integer Uniform, Binomial, Gaussian, Log–Normal, arbitrary PDF or CDF, etc. For the implementation in FireGrid, two PDFs were used, i.e. arbitrary PDFs for parameters such as fire scenarios, family profiles, fire origin, etc, which are adequately described by a relatively limited number of states, and Gaussian PDFs for properties of burning items and others where it is

much more challenging to properly resolve the distributions using a finite number of sample bins.

As the arbitrary PDF is discrete, with the options (histogram bins) predetermined and unchanging, Bayesian inference simply determines the relative probabilities for each option. After every sensor update, the weight of each scenario is accumulated according to its selected parameter value, i.e. which "bin" it falls into, to form a likelihood function, L_{bin}; then, the posterior PDF can be derived by applying the likelihood function to the prior PDF, i.e. $p_{posterior}(bin) = p_{prior}(bin) \times L_{bin}$; where *bin* is the option that parameter *p* has selected, $p_{posterior}$ and p_{prior} are the posterior and the prior probabilities, respectively. As the distribution is completely arbitrary, it cannot be smoothed, but it must be ensured that none of the probabilities go to zero (otherwise future updates will never change from zero).

Unlike the arbitrary PDF, the Gaussian distribution is continuous. In principle, it could be represented by a histogram in the same way as the arbitrary PDF (each histogram bin represents a value between x and $x+dx$). It is clear that no matter how many scenarios are simulated, one would end up with a finite number of points in the likelihood function rather than a smooth line; thus it may lead to a very "spiky" posterior function if Bayesian inference is used. However, if it is insisted that the posterior probability distribution is also a Gaussian, a simple solution presents itself.

Using Equations 15 and 17, and the further simplification that $N(A_j)$ can only be 0 or 1 due to the infinitesimal histogram bin widths, the mean of the posterior Gaussian distribution is

$$\mu(x \mid B) = \frac{\sum_{k=1}^{N_{tot}} \alpha_k P(x = x_k) x_k}{\sum_{n=1}^{N_{tot}} \alpha_n / N_{tot}} \tag{18}$$

and the variance is

$$\sigma^2(x \mid B) = \frac{\sum_{k=1}^{N_{tot}} \alpha_k P(x = x_k) x_k^2}{\sum_{n=1}^{N_{tot}} \alpha_n / N_{tot}} - \mu^2(x \mid B) \tag{19}$$

where x_k is the value of the parameter value x in the k^{th} scenario, α_k is the goodness–of–fit of the k^{th} scenario, etc.

In the previous section it was stated that, providing certain conditions were met, the quantity X^2 defined in Equation 10 would follow a x^2 distribution with an appropriate number of degrees of freedom. The properties of the x^2 distribution were then used to determine the goodness–of–fit of each modelled scenario. However, if in reality these conditions are violated, then a procedure to maintain a reasonable number of matched scenarios at every update will be required. Hence, a simple scaling factor has been adopted for X^2, which can be slightly increased or decreased at each time–step based on the overall assessed goodness–of–fit, thereby maintaining the sum of the weighting factors at about 10%~20% of the number of simulations N_{tot}, Thus, if the model has produced enough scenarios, the factor is reduced so that the model can focus on more accurate scenarios, and *vice versa*.

3.5.3 The concept of fire "phase"

In application of the above to steering in K–CRISP there are also some further practical issues which require further consideration for efficient running of the sensor–linked model. It is noted that in the K–CRISP representation, different aspects of the fire behaviour are most closely related to specific input parameters; for example, rate of the flame spread is one parameter directly influencing the gradient of the temperature rise, while initial fuel load has a direct effect on the point in time when the fire goes into the decay phase. Randomizing all

of the parameters throughout the simulation, therefore, may not be an optimum approach. Moreover, for a relatively simple 4–room enclosure (including all the parameters related to fire, building and occupants), such as that considered in the full–scale test of FireGrid D7.4 demonstrator, K–CRISP currently randomizes a total of more than 100 parameters. Hence, for most practical cases it would be necessary to have a huge number of runs in order to even approximately cover all the possible scenarios, resulting in computational limitations when required for real–time applications.

The concept of fire "phases" was therefore introduced in order to facilitate a more efficient approach to the problem. These phases are defined as "Initiation", "Growing", "Non–growing" and "Full–room", and each fire parameter is categorized into one or more of these different groups, based on the specific phases it influences. The purpose of the *initiation* phase is to allow the model to infer information from the sensor data before it starts comparing it to simulation outputs and also to handle the potentially sudden transition in the fire conditions which occurs at ignition. Parameters such as fire origin and ambient temperature were included in this phase because these are important early on, since they can improve the efficiency of the simulation if known but they can make the output irrelevant if badly wrong. The *growing* phase includes rate of flame spread, latent heat of vaporization, item height and maximum radius of burning item. Parameters such as fuel load and exponential decay time, which affect the fire only after it enters the decay stage, are in the *non–growing* phase. Once the fire goes beyond the first item it is assumed to involve the entire room and the *full–room* phase parameters, which are basically properties of the item "full–room", govern the output. The sensor–linked model will check the measurements at every update and based on the current temperature, the slope of the temperature rise, etc., it will make an assessment of the current fire phase and the relevant parameters will be retained as fully flexible or fixed, accordingly; other more general parameters such as thermal properties of the compartment boundaries are retained as variables throughout.

3.6 Sensor data

All sensor measurement data is passed to the fire model via a database, as per the architecture developed for the FireGrid project [1]. Formatting conventions are adopted for each sensor type, enabling rapid set–up. The detailed test specifications and measurement data are archived under FireGrid [31]. Tools have also been developed to allow virtual replay of sensor measurements via reading of the database, thus permitting study of model performance to continue at leisure after any live event, such as the full–scale fire test studied in this thesis.

Some thought needs to be given to the possibilities of sensor failure and the need for live screening of the measurements. In general, sensors such as thermocouples are robust enough to survive in high temperature environments, up to 1000°C and beyond. However, failure is not unknown and such cases may lead to the generation of erratic or spurious data; thus, if an adequate filter is absent, the whole steering process might be affected and misled, thus resulting in inaccurate predictions. It is therefore critical to detect the failure of a sensor as soon as possible and block or filter the data so that the updating procedure can ignore it in time. There have been a number of previous studies on filtering out false sensor data, especially in the signal processing field, and sophisticated methods have been devised. But since the main concern of relevance to the FireGrid project is simple “out–of–range” values from sensors, as they are potentially more serious in terms of their impact on the steering procedure, it was elected initially just to implement a simple filter in the model, i.e. each sensor channel was given fixed upper and lower bound values and if the data was found to lie outside this range it was considered as a “failure” and automatically excluded.

It is useful to recognise that sensor measurements can provide useful direct information about the fire and the parameters being adopted in the model. For example, if the status of a certain door (open or closed), or even which specific items are involved in the combustion

process, is known via dedicated measurement technologies such as door closure sensors or video images, this information can be fed into the model before the updating is carried out to "predict" it. This can help in lowering the number of randomized parameters and it will tend to increase the accuracy of the model. For this reason, the model will check sensor data and derive all possible direct information first, bypassing the updating for those parameters, as appropriate. The information includes not only the parameter values themselves, such as ambient temperature, fire origin and window breakage, etc., but also the assessment of the current status of fire, i.e. fire phase, which will then be used to release or restrain the parameters based on their category aforementioned.

3.7 Hazard prediction

If K–CRISP model is run beyond the time of the current sensor observations, it can in theory be used to provide some estimate of the future development of the scenario – not just further sensor measurements, but also, for example, how fast the fire is growing, how likely flashover is to occur, whether people will be exposed to smoke and how likely this is to prove fatal, whether there will be failure of structural components, etc. The range of variables that can be predicted obviously depends on the capabilities of the model being used for the task. In K–CRISP, the fire model was modified to output the hot layer depth and temperature in each room, the number of people and their Fractional Effective Dose (FED) of toxic species (e.g. CO), combustion products (i.e. CO_2), and hazardous exposures (heat) in each room, structural deflections and "structural integrity" parameters for specific members, as well as a complete set of the predicted sensor data, for some reasonable time into the future.

For the potential end–users such as fire–fighters, building operators, etc., however, raw output consisting of numbers of data lines representing different fire scenarios might not be

useful at all. In an emergency situation, a simple way of illustrating the results is required so that the information can be delivered intuitively. In the FireGrid project, the user interface for the end–user was also developed exploiting procedures from the field of artificial intelligence [40], in collaboration with input from fire–fighters [27][41]. The user interface aspects are beyond the scope of this thesis but the essential content of the information provided by K–CRISP is outlined below.

After every sensor update K–CRISP will carry out updating and produce a summarised prediction output for all future time points of interest. The prediction data at each time consists of two parts. In the first part, all the values in each scenario are averaged after being multiplied by a weighting factor, which is the "average of weighted predictions". This can describe the overall trend of the fire development in a single line, without losing its relationship to the actual fire and retaining knowledge of the uncertainty in this estimate, via the X^2 value. The second part is the extracted or derived information which the model will determine to deduce the probability of having a certain "risk" by examining all the scenarios generated. In other words, these parameters can provide answers to questions such as: "How likely is it to have a temperature over 500°C?". In this example, the point in time when the temperature exceeds 500°C is marked for each case by first investigating all the scenarios. Then, the cumulative probability is calculated by counting those scenarios and rearranging them in the order of time. As a result, the end user can be supplied with outputs such as: "In 50% of generated scenarios, the temperature exceeds 500°C at 315 seconds from detection".

Finally, it is very important to recognise that with a lack of any specific representation of the orientation of individual combustible items, as is conventional in zone models, there will be no inherent capacity to predict sudden changes in fire behaviour associated with rapid flame spread or remote ignition. The same is true of a number of other phenomena occurring in real fires, e.g. explosions, various types of compartmentation failure (including those due to

burn–through or glazing failure), fire growth due to collapse of burning items, various types of human intervention, external wind factors, etc. Though the modelling procedure allows the possibility of most of these occurring, e.g. a probabilistic approach is used to describe glazing failure, and loss of integrity may be inferred after the event, no claim can be made concerning predictive capability in such phenomena. Thus, when conveying the results to the end–user, appropriate allowances would need to be applied, over and above the basic inferences on likely fire development, to consider any unknown hazards which might often be present in real fires.

3.8 Testing sensor-linked fire model with live sensor data

Two Fortran–based utilities, i.e. PreProcessor and FireSelector, were embedded in K–CRISP in order to test the model in pseudo live mode and to carry out the steering procedure without affecting the performance of generating scenarios.

The PreProcessor simulates precisely the same environment for K–CRISP as existed in the live fire tests. It simply reads all the sensor measurements but only sends them to the simulation component of K–CRISP up to the current point in time, i.e. pseudo real–time. The FireSelector is a unit which carries out most of the ‘analysis’ procedures such as ‘matching’, ‘evaluating’, and ‘steering’. Once the PreProcessor sends new sensor data, the FireSelector first analyses sensor observations, then decides which phase the fire is in, reviews the prior PDFs being used, evaluates scenarios based on statistic criterion, carries out Bayesian inference calculations, and finally sets posterior PDFs in which new scenarios are generated.

Although K–CRISP consists of multiple units, i.e. CRISP, the PreProcessor, and the FireSelector, it was designed in a way that each unit is executed independently and communicates with each other via “files” rather than being a single combined program

which may then be able to “talk” to other units via data held in memory. Being a single program provides the advantage of quick communications between the units, as it does not spend much time to save data in files; however, the model can potentially have a bigger risk in terms of guaranteeing a sustainable and stable run. As CRISP, the fire model in K–CRISP, carries out relatively more complicated calculations than other units, it has a bigger chance to become stuck and thus to stop generating scenarios. This could then potentially cause the whole of the K–CRISP system to malfunction if a sophisticated programming method for multi–processing is not adopted.

Separating each unit, on the other hand, may lead to the model spending more time to save and read data via files; however, it is more flexible in terms of parallel computing and easier to deal with aforementioned malfunctioning. While CRISP heavily uses the computational resource all the time, the PreProcessor and the FireSelector use relatively smaller resource, as they only have to “calculate” when a new set of sensor observations is fed in. Thus, it is expected that multiple CRISP runs have to be performed in parallel to generate sufficient numbers of scenarios but only a single (or a few) units of the PreProcessor and the FireSelector are needed to handle those outputs. The model just needs to be executed in multiple PCs or processors of HPC as required. Moreover, even if a single CRISP unit stops generating scenarios, it will not affect other units and outputs from a specific instance of CRISP can easily be excluded from the entire system during the run.

One challenge for the real–time emergency response system is the need to enable the model, K–CRISP, to simulate a sufficiently large numbers of scenarios in short time, e.g. 30 seconds, in order to identify parameter values giving good matches to the observations. Therefore, K–CRISP was developed in Fortran environment enabling easy conversion between Windows–based PC and HPC resources such as the ECDF machine of the Edinburgh Parallel

Computing Centre (EPCC), known as Eddie, and the national HPCx machine [31][41], using Linux and IBM AIX operating systems, respectively.

Of particular note for the FireGrid system is the requirement for access to the HPC resource in “urgent computing” mode. This is necessary to ensure that K–CRISP can be launched as quickly as practically possible after the time of request, and irrespective of other computational loads on the resource. For ECDF, this requirement was fulfilled by isolating a portion of the machine (a single, 8–core compute node) from the main cluster and deploying a separate instance of the batch system, names Sun Grid Engine, to manage it as a dedicated resource for FireGrid system jobs. For HPCx, an allocation of compute power (one 32–core compute node) was reserved prior to each experimental run, via an advanced reservation request to the general HPCx batch system, named LoadLeveler. The reservation effectively provided a dedicated compute resource for the project team during each fire experiment.

Prior to the FireGrid D7.4 demonstrator, K–CRISP was ported to multiple PCs linked via LAN network in order to carry out the test even with unexpected failure of HPC resources. With sufficient numbers of PCs, it was possible to build a similar environment to that provided by the HPC resources without any code modification. Further investigations of model performance were also undertaken via virtual replay of the measured values, i.e. in pseudo real–time. While a pseudo live environment is achieved by the PreProcessor, i.e. the sensor data will be fed into the model exactly the same way as it had been in the live test, pseudo real–time is achieved by updating sensor data only when a certain number of scenarios are generated. This was to allow K–CRISP to achieve an equivalent amount of simulations, even with a single laptop PC, as with the HPC resources or multiple PCs. All the results of K–CRISP predictions shown in the following chapter were generated in pseudo real–time.

Chapter 4

Application of sensor-linked fire model

4.1 Introduction

A sensor–linked model was developed based on the methodology described in the previous chapter. The model successfully played a key role in the FireGrid prototype emergency response system as the ‘Simulation Component’ integrating with other units such as the ‘Agent–based Command–Control Component’ and the ‘Data Acquisition and Storage Component’. In this chapter, the sensor–linked model, K–CRISP, is demonstrated using pseudo live fire data obtained from the two full–scale fire experiments, the Dalmarnock fire tests and the FireGrid D7.4 demonstrator, and the model’s predictions are assessed in detail from various aspects of compartment fire.

The Dalmarnock fire tests were carried out in relatively early stage of the research in this thesis. Thus they offered a chance to test the model during its developing stage in terms of the performance of the fire model in K–CRISP, assessment of the evaluation criterion, and optimizing the 'randomized' items (see section 4.2). The predictive capability at this stage was then assessed using the best–match scenario based on the scenario selecting procedure and the results are shown in section 4.3.

Although a large number of sensors were installed in the compartment of the Dalmarnock fire test, they were mainly focused on the room of fire origin and few sensors were installed in other rooms in the flat. In the FireGrid D7.4 demonstrator on the other hand, sensors were similarly installed in all rooms in terms of their locations, sensor types, and density because the main purpose of the test was to assess the performance of the FireGrid prototype system rather than to accumulate detailed information of fire behaviour in the compartment. Therefore, the test result was more appropriate as a basis for examining the sensor–linked model, as it can offer more options in terms of which sensors are (and should be) used in the steering procedure. For this reason, the steering aspect of the sensor–linked model was tested using the FireGrid D7.4 demonstrator data and the results are shown in section 4.4.

Finally, the predictive capability of K–CRISP is assessed using the Dalmarnock fire test results with the same model parameters used in the FireGrid D7.4 demonstrator (section 4.5). This is to run the model truly blind and to be assured that K–CRISP can produce similar results for the specific test without any knowledge of the fire.

4.2 Model verification using Dalmarnock fire test data

Several tests were first carried out using the Dalmarnock fire test results [4] in order to verify that the model can reproduce a realistic fire scenario with reasonable accuracy, that it has

sufficient flexibility with its randomized input parameters, and finally that it can generate reasonable predictions with a simple matching procedure for this type of fire. For the testing purpose, the standard deviation was used as a matching criterion and the best–match scenario based on the scenario selecting procedure was used. The predictive capability based on a live "steering" procedure based on this method of scenario selection is shown in a later section (section 4.4).

In the Dalmarnock fire test, the burning of the waste–paper basket, the blanket and the cushions, and the sofa controlled the initial fire development. The sofa's heat release rate in freely ventilated burning had been measured in advance in the laboratory and heat release rate after the onset of the ventilation–controlled phase was estimated based on the assumption of oxygen depletion using velocity probe measurements. Figure 4.1 shows the average temperature of the Dalmarnock fire test compartment, one measured from the live test and the other reproduced by K–CRISP using this heat release rate data as input. Simple visual inspection of the figure reveals the match is a good one; certainly the prediction lies well within the boundary of maximum and minimum temperatures of the Dalmarnock test. The average temperature was under–predicted until 150 seconds from ignition; however, it showed a good match thereafter. It is therefore fair to assume that K–CRISP can generate the fire conditions with sufficient accuracy for this type of scenario from the point of view of average compartment temperature, and that it will be able to select a "reasonable" match via Monte–Carlo simulations assuming the K–CRISP fire model is sufficiently flexible.

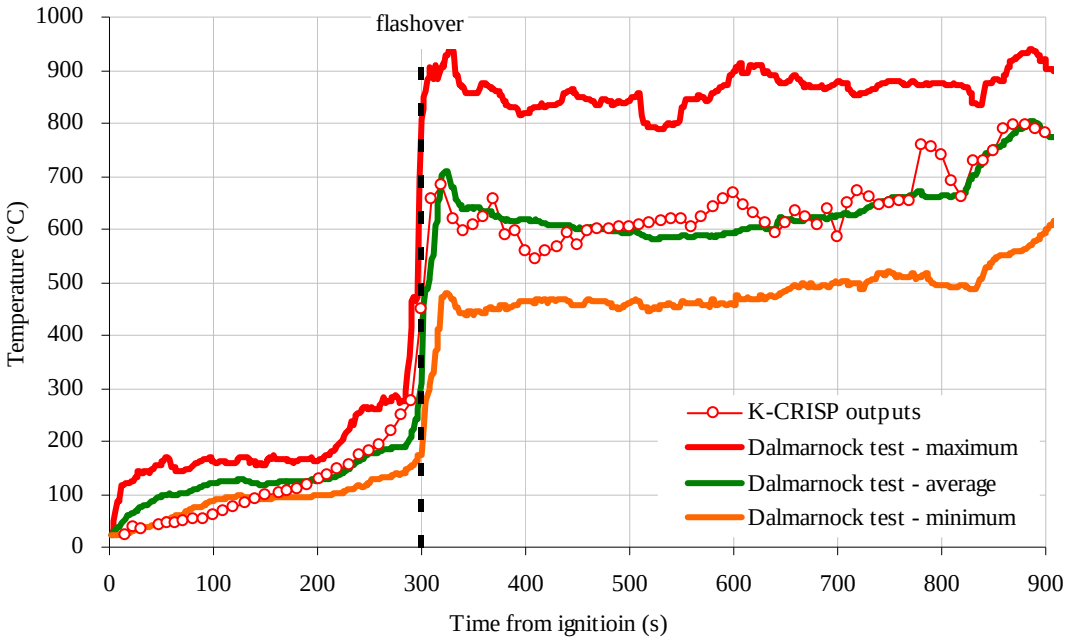


Figure 4.1 – Dalmarnock Fire Test One data and K–CRISP prediction using estimated HRR

A single burning item was initially used to fit the Dalmarnock fire test data, however, the test observations (fairly steady temperature up to 300 seconds, followed by a rapid rise and then fairly steady again) were not well represented by this model. A better fit was obtained by allowing a second item to ignite some time after the first, with the parameters of the second item sampled from the same PDFs as the first item. Figure 4.2 presents the best–match scenario after two sets of 1000 simulations using randomized input data, one with a single item and the other with two items. Both the best single–item and two–item scenarios give a good fit to the steady–state temperature conditions after 300 seconds. However the single–item scenario struggles to match the growth phase; initially the growth rate is too slow, but after 150 seconds from ignition the predicted temperature exceeds the observed value, in order to reach the final peak value reasonably soon after the observed temperature stabilises. With the two–item scenario, a much better match is seen, and the first of the two items is burning out (as evidenced by the decline in temperature between 150~270 seconds) before the second item ignites.

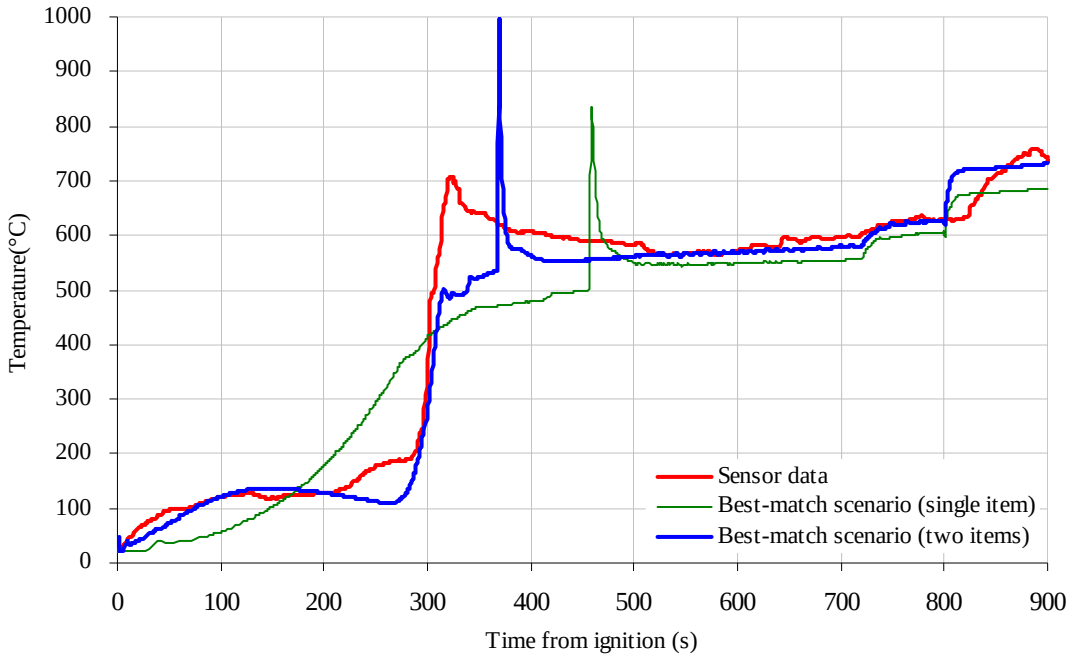


Figure 4.2 – Comparison of the Dalmarnock fire test data and overall best–match scenario

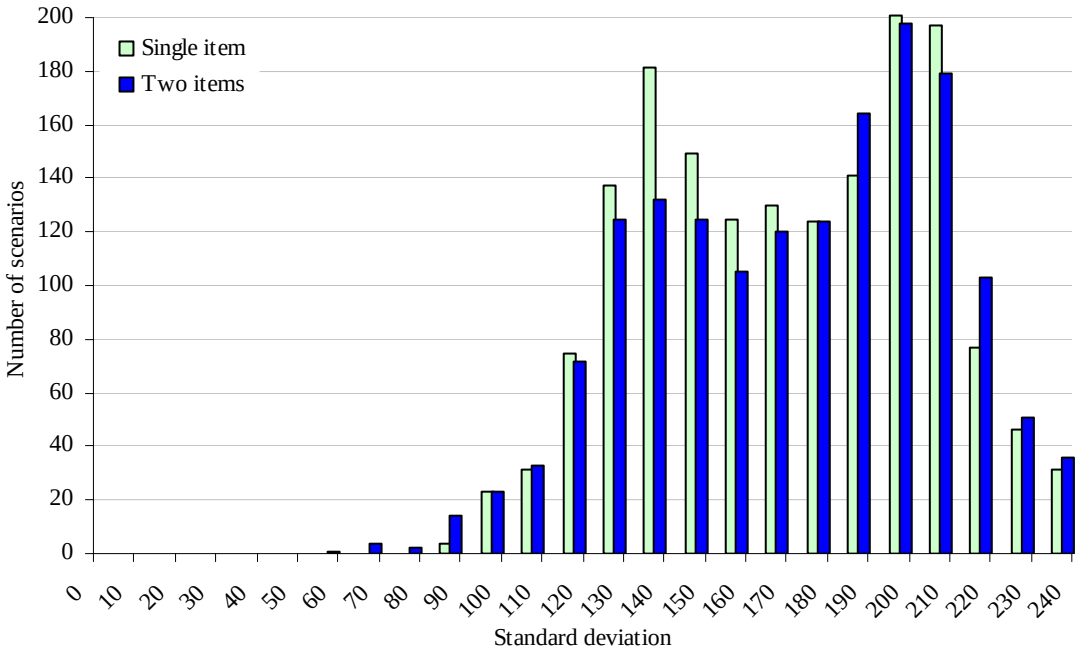


Figure 4.3 – Distribution of standard deviation using different number of burning item

Figure 4.3 shows the distribution of standard deviation obtained from 2000 Monte–Carlo simulations, half with one burning item, the remainder with two items. The latter were

capable of giving a better fit with the right combination of parameter values. While the best standard deviation using a single item was 88, it was 56 when two items were used. When the estimated Dalmarnock heat release rate curve was used as input, it could be lowered to 48, as the model simply generates temperatures based on the overall heat release rate, i.e. decoupled from the number of burning items.

Figure 4.4 shows 100 randomly selected K–CRISP scenarios (full range of calculated temperature–time curves) from among 2000 Monte–Carlo simulations. The thick red line represents the average compartment temperature measured from the Dalmarnock fire test. It is interesting that most of the scenarios do not get up to temperatures beyond 600°C. This may be because the fire is restricted by the ventilation limit during this phase and the temperature cannot be higher even if sufficient fuel has been supplied. This is also consistent with the fact that the temperature slightly rose at about 720 seconds when the kitchen window was broken and at 801 seconds when the living room window was broken (see Table 2.1, page 19) supplying more oxygen to the fire compartment.

It is clear that most curves do not match the Dalmarnock fire test temperature curve very well, although a few do. This is not surprising, because we allowed large variances for our PDFs of parameter values. Also, the scenario generation adopted here is fully “blind”, i.e. involves no feedback, and hence there is no learning about the fire scenario; when live steering is adopted the scenarios most resembling the current best match can be generated disproportionately, as described later (section 4.4). Note that the short–lived “spikes” in the figure predicted by K–CRISP are a consequence of the flashover sub–model. Because they are short–lived, the effect of the spikes on the matching criterion is relatively small.

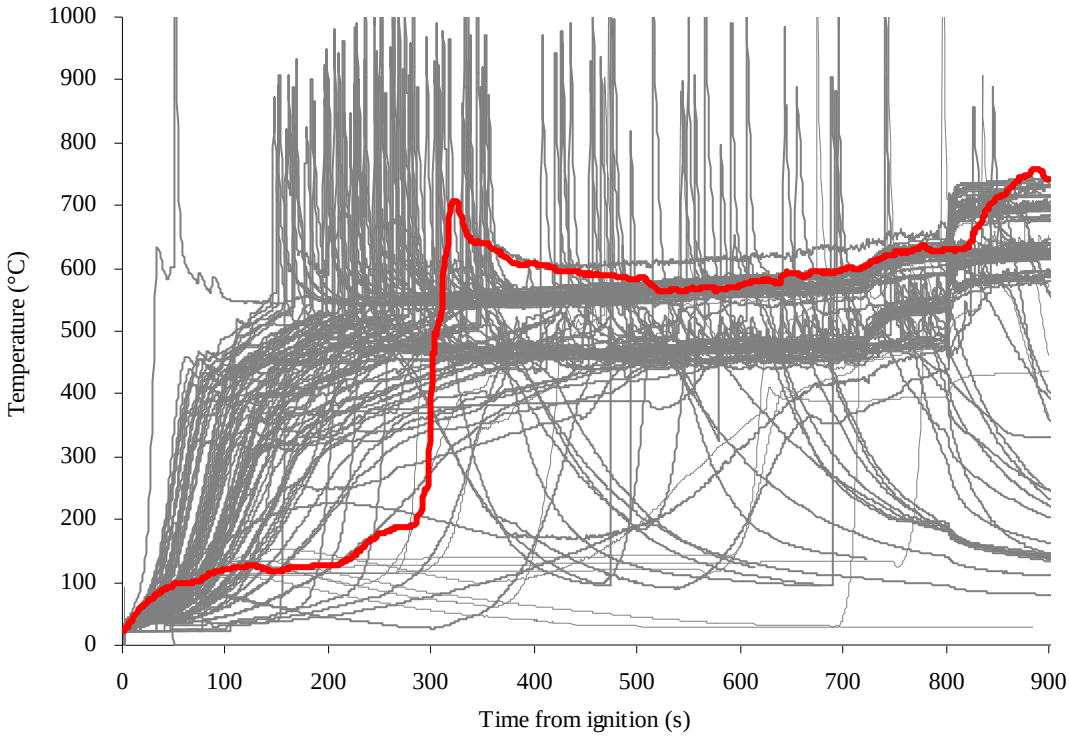


Figure 4.4 – 100 selected K–CRISP outputs among 2000 simulations

As shown in the figure, the sensor observation lies inside the range of various scenarios, therefore, it can be assumed that the randomized input parameters enables K–CRISP to be sufficiently flexible so that it can generate multiplicity of scenarios which can possibly cover the “real” fire (relevant to this specific scenario).

The test results shown in this section confirm that K–CRISP is capable of reproducing this particular fire scenario in terms of average compartment temperature with sufficient accuracy. Randomized properties for input parameters, such as those related to the burning item, have successfully supplied the model to allow creation of various fire scenarios. Therefore they are expected to cover most of the possible fire scenarios in the compartment if sufficient numbers of scenarios are generated. Finally, the use of a criterion based on standard deviation enabled K–CRISP to select a best–match scenario which gives a sufficiently good fit to the real fire. However, it also showed the importance of the secondary

(or multiple) burning items, which is essential to reproduce the fire in a realistic compartment more accurately.

4.3 Forecasting fire development using scenario selecting procedure

As described above, K–CRISP was initially tested using compartment fire test data and the performance was satisfactory in terms of its accuracy and flexibility. In work described in this section, K–CRISP was run in pseudo real–time and the predictive capability of K–CRISP based on the scenario selecting procedure is presented.

K–CRISP was ported to a laptop PC (2.0 GHz, 2 GB RAM, under Windows) and the Dalmarnock fire test results were used as pseudo live data. K–CRISP performed simulations much faster than real–time. Even with a regular PC environment, it was possible to analyse 1000 cases in less then 20 minutes. On an individual HPC processor (2.6 GHz, 2 GB RAM, under Linux), the speed of execution is nearly identical, but of course the Monte–Carlo approach is ideally suited to the Grid/HPC as it just needs to run on as many processors as are available and collate the results [42].

As mentioned in the previous chapter, the PreProcessor, an embedded unit in K–CRISP, only sends sensor data to the simulation component in order to run the model in pseudo live mode. K–CRISP will select the best–match scenario based on the standard deviation calculation using sensor observations up to the ‘current’ point. In this live test, it was assumed that sensor data is fed in every 10 seconds and K–CRISP will carry out the selecting procedure and update its selection if a better scenario is found.

Figures 4.5 to 4.8 show the updates of the choice of the current “best” case at four different times. The vertical black thick line indicates the “current” instant in time and the sensor

measurements are only “known” up to this point, as indicated by the thick red line, with the dotted portion showing the future trends. The blue line represents the best matched case.

At 30 seconds from ignition, the best–match case was scenario #309; however, this scenario predicts a very rapid growth which is not supported by the subsequent sensor observations. The choice of best scenario is therefore continually varying as more observations become available. At 290 seconds (see Figure 4.7), the sharp increase in temperature causes selection of scenario #141 which has a similar temperature increase. At 300 seconds (see Figure 4.8), the temperature starts to peak and scenario #714 is selected since it has a bigger temperature jump at around 300 seconds. In spite of the gap between the results of Dalmarnock test and the selected CRISP scenario around the flashover period, this remained the best–matched scenario among all the cases even as more sensor data became available later.

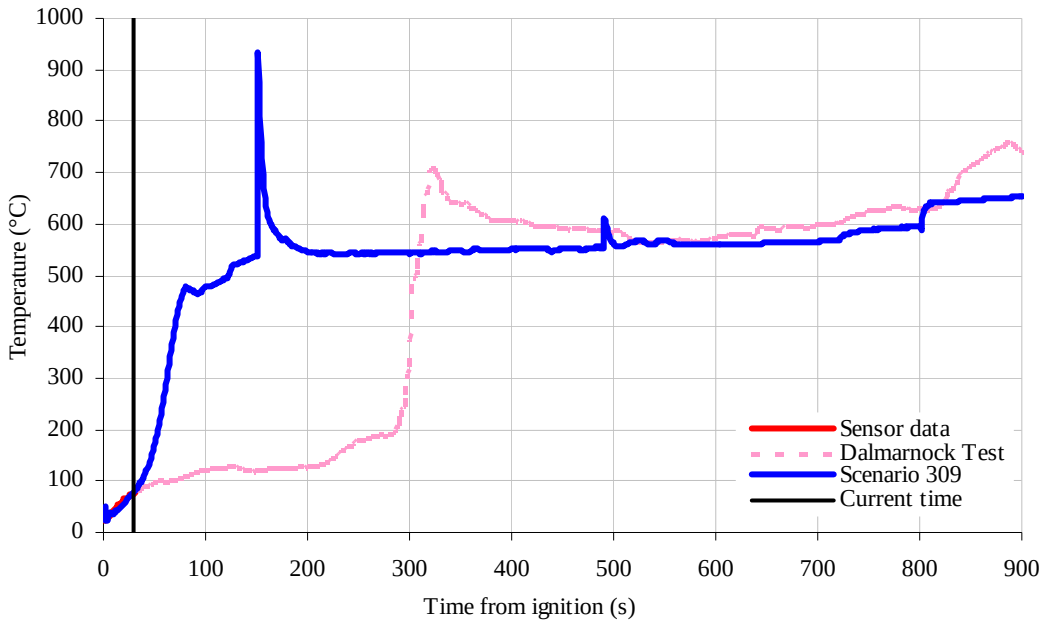


Figure 4.5 – Real–time prediction at 30 s from ignition based on scenario selecting

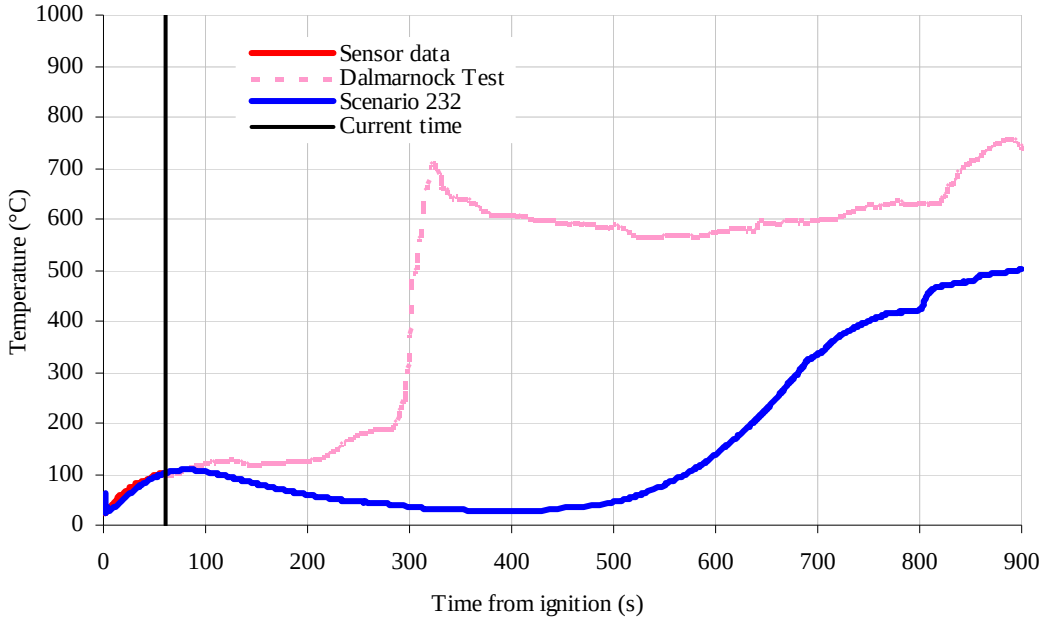


Figure 4.6 – Real–time prediction at 60 s from ignition based on scenario selecting

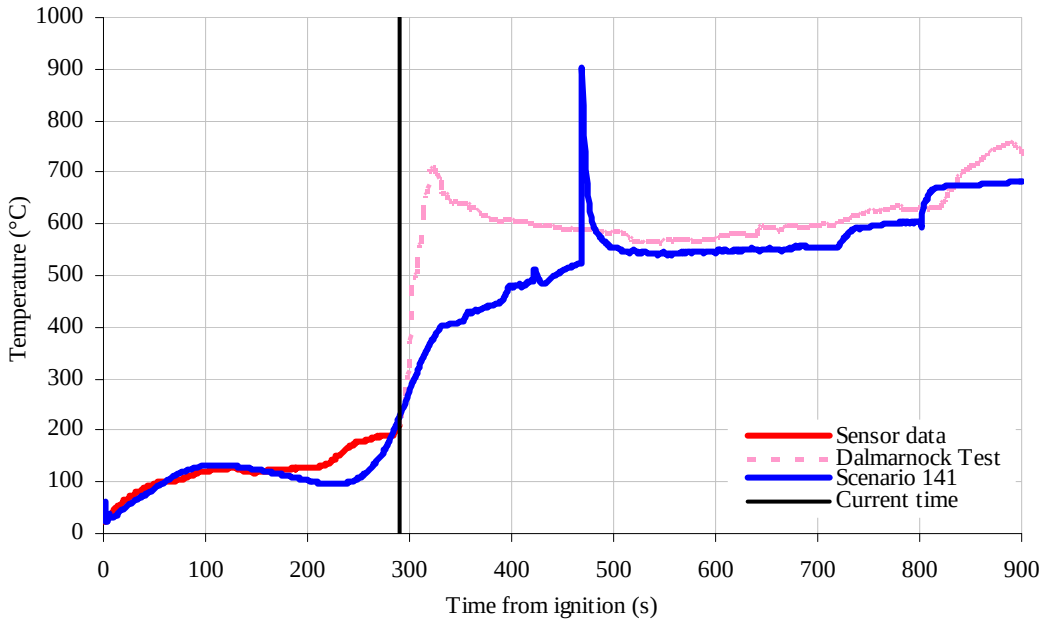


Figure 4.7 – Real–time prediction at 290 s from ignition based on scenario selecting

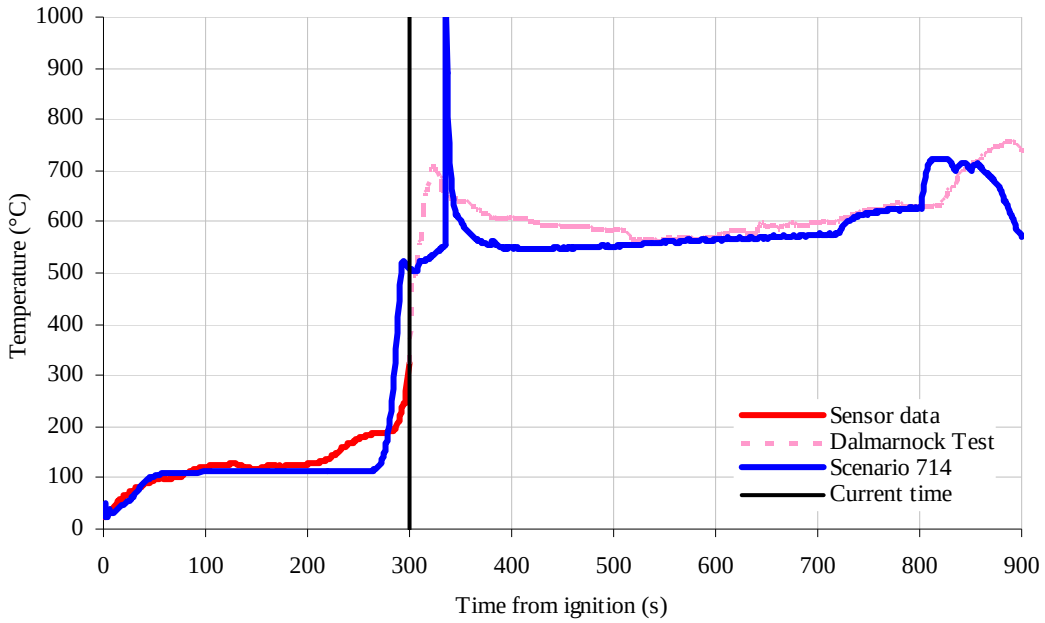


Figure 4.8 – Real–time prediction at 300 s from ignition based on scenario selecting

The K–CRISP results have shown that the scenario selecting procedure is conceptually simple and capable of providing super–real–time forecasts. However, it was seen that the model kept generating irrelevant scenarios even after some degree of information of the fire has been revealed by the sensor observations as it is generating new scenarios independently of the selection process. Let us consider that sampling of new scenarios could be concentrated on the neighbourhood of the optimal scenarios. This ought to lead to a narrowing of possible scenarios, i.e. reduction of the variance of the ensemble as more sensor data becomes available. It was also seen that the scenario may not be capable of resolving critical events like flashover, even when the standard deviation between the selected optimal scenario and the data is small. This confirms that a single best–matched scenario may not correctly describe the overall trend of the fire development in other well–matched (but not the best–matched) scenarios and a sophisticated way of summarising the model outputs, i.e. average weighted prediction, is required. Considering these limitations of a methodology based solely on scenario selection we now proceed to examine the further development of the approach to include model steering.

4.4 Forecasting fire development using model steering procedure

K–CRISP was capable of forecasting fire development even with the simple scenario selecting procedure using a laptop PC. Although some level of prediction of fire was achieved, the efficiency of creating new scenarios can in principle be greatly improved by using feedback loop between the scenario assessment and the parametric space for generation of new inputs. Here we examine the performance of this type of approach (as described in section 3.5) in application to the FireGrid D7.4 demonstrator experiment.

4.4.1 Performance of model in HPC

During the FireGrid D7.4 demonstrator, the simulations of the fire scenario using K–CRISP were performed remotely on HPC resources. Since the underlying modelling approach is derived from the Monte–Carlo concept, which involves generation of large numbers of independent scenarios, it is ideally suited to implementation on multi–processor machines. K–CRISP could carry out simulations of the case much faster than real–time using only a few processors on the HPC resource. It was possible to produce more than 1000 scenarios per minute, to perform Bayesian inference, and to come up with posterior PDFs every 30 seconds, which was the sensor data update frequency. It ought to be feasible to model more complex scenarios, or longer duration runs, simply by calling on greater amounts of HPC resource, and also to increase the frequency of the updates as required. Subsequent to the test, further investigations of model performance were undertaken via virtual replay of the measured values, i.e. in pseudo real–time. This allowed careful exploration of model performance, decoupled from any computational constraints. The results presented in this section were obtained from a pseudo real–time run. Nevertheless, for comparative purposes, all results presented here relate to approximately the same number of scenarios per update as was considered in the live test.

4.4.2 Assessing model prediction of average compartment temperature

In the FireGrid D7.4 demonstrator, the test rig was heavily equipped with various types of sensors such as thermocouples, velocity meters, and radiometers. K–CRISP can also simulate most of those sensor types; however, careful thought should be made to decide what sensor data is to be used in the steering process. Firstly, not all kinds of sensors can be supplied in a real building. Heat release rate, for example, is frequently used as a criterion to define fire development or to compare two different fires. Although no buildings have calorimetry infrastructure, it is possible to make rough estimates of the total heat release based on measured flows and the principles of oxygen depletion calorimetry, assuming that all of the oxygen in the inflow air is consumed [29]. However, such attempts will be confounded by uncertainties, in particular regarding the location of the burning, much of which may occur externally, Temperature sensors, on the other hand, are widely used for various purposes and can more naturally characterise the fire conditions in the compartment; however, realistic buildings may have only one (or none) of them installed in each room, unlike the condition of the live test where numbers of thermocouple trees were installed which can measure temperatures at different heights. Secondly, not all simulated sensor data in the model's outputs are expected to be sufficiently accurate to be used in the steering procedure due to the conventional limits of the model being used (i.e. the typical approximations of a zone model, with only two uniform gas layers).

As described in the earlier chapter, parameters such as fire origin and door status (open or closed) were preset during the live test. The fire was planned to be ignited from the sofa in the *Room 1* and most of the potential burning items in the compartment were well characterised. All the doors were also set to be fully opened in the test in order to supply enough air to the fire source. In the real fire case, however, it is clear that such information

about the burning items or the door status may not be available at all and it may be difficult to figure out even where the fire has started.

A “default” case was therefore defined in order to be assured that the model is capable of generating predictions using simple characteristic inputs such as temperatures, but also to make sure it can “predict” information such as fire origin and door status which are not likely to be given to the model in the real fire case even though they are very essential in terms of accurate predictions. The average compartment temperature of each room was chosen as the initial “sensor observation” to steer K–CRISP and they were all given the same scaling factors considering the fact that any room can potentially be the fire origin. The *fire origin* parameter was excluded from the fire *phase* so that it is inferred by the steering process, not by the direct bypassed information from a fire alarm. Additionally, the status of all doors was fully randomised, i.e. each door was initially given 50% chance of being open.

A concept of “average of weighted” predictions was used to summarize all the hot layer temperatures as well as other outputs such as hot layer depth and temperature in each room, the number of people and their Fractional Effective Dose (FED) of toxic species such as CO and hazardous exposures (heat) in each room, structural deflections and “structural integrity” parameters for specific members.

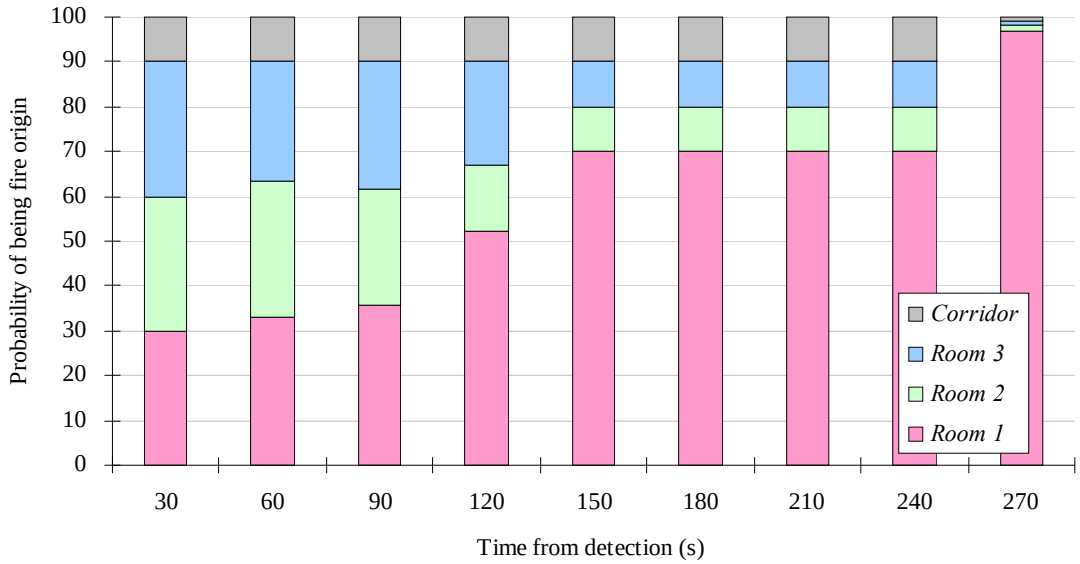


Figure 4.9 – Change of PDF for *fire origin* of default case

Figure 4.9 shows the change of the PDF for the *fire origin* parameter. Each room, *Room 1*, *Room 2*, and *Room 3*, was initially given 30% chance of being the fire origin and 10% to the *Corridor*. As more sensor observations are fed into the model, *Room 1*'s probability of being fire origin gets bigger and eventually becomes 70% while other rooms are set to 10%. Note that the probabilities of arbitrary PDFs for fire origin were set to have minimum 10% as the future updates will never be able to change if it becomes 0%, as indicated in the previous chapter. But once the average compartment temperature of the fire origin room is high enough compared to those of other rooms, the model 'decides' to lower the minimum probability to 1%, as shown in the last column (at 270 seconds) in the figure.

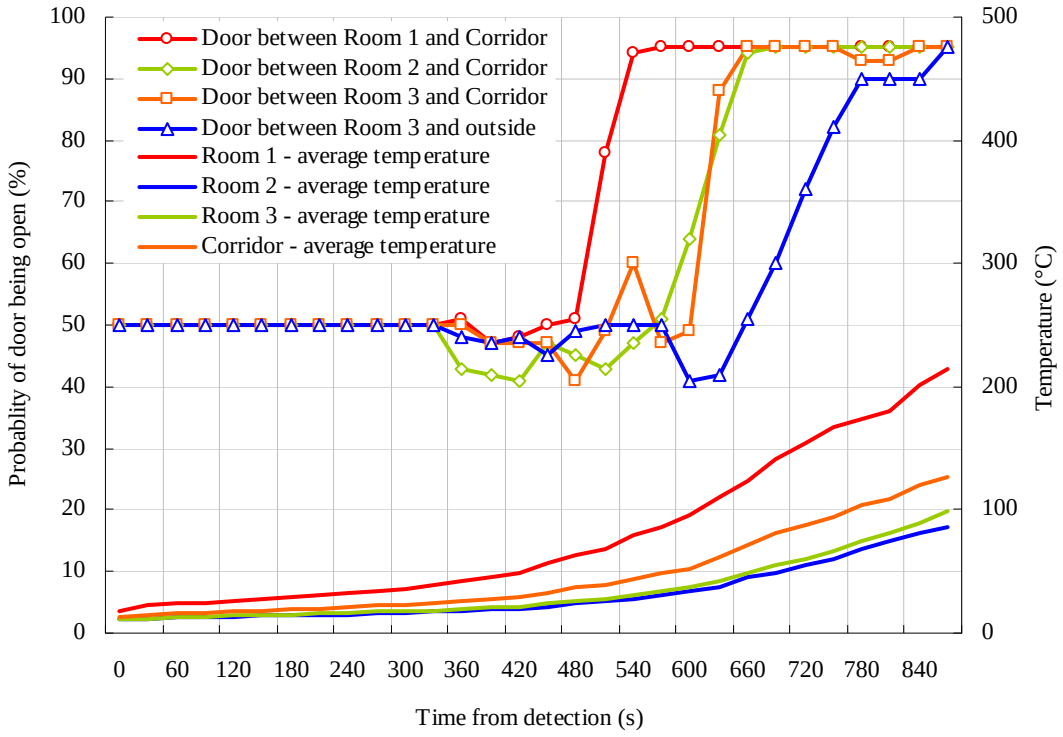


Figure 4.10 – Change of PDF for *door status* of default case

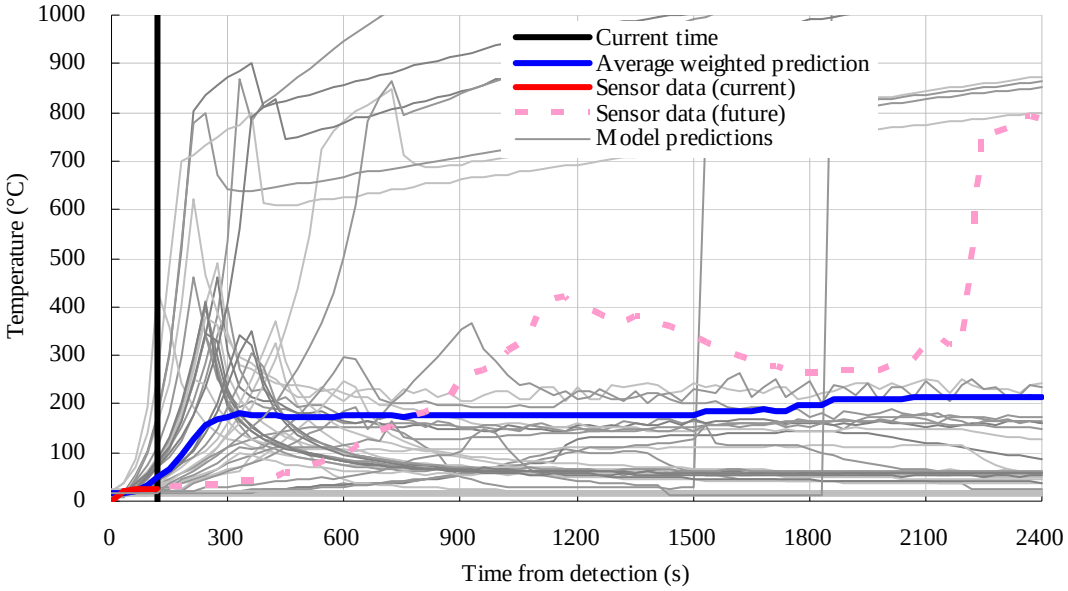
Once the fire origin is 'known', K–CRISP then tries to figure out which door is open and which is not. It was intended that the model does not change PDFs for door status until it deduces the fire origin, as it is possible that it might converge the PDFs for doors too early while the fire origin is not yet inferred. Figure 4.10 shows the change of the probability of each door being open during the early stage of the fire development with the average compartment temperature of four rooms plotted below. After 300 seconds from detection, the PDFs for the doors were released as the fire origin was inferred as shown in the previous figure. While the PDF for the door between *Room 1* and the *Corridor* converges quickly to "open", K–CRISP struggles to infer the status of the other doors. After 600 seconds from detection, another two doors in the *Corridor* converges and finally the PDF for the exterior door in *Room 3* becomes 95% (the minimum probability of the door being closed was set to 5%). This result indicates an interesting fact, that K–CRISP can only infer "door is open" when the temperature of two rooms at both sides of the specific door is high enough. In other

words, if there is no change in one of the rooms, the model will assume that the door is closed, thus no hot gas is coming out from the fire room. This also explains why the PDF for the exterior door in *Room 3* was converged last, because no temperature data was given from the "room" outside. Thus the model has to infer it from the overall compartment temperatures not from the comparison of two temperatures, one from *Room 3* and the other from outside.

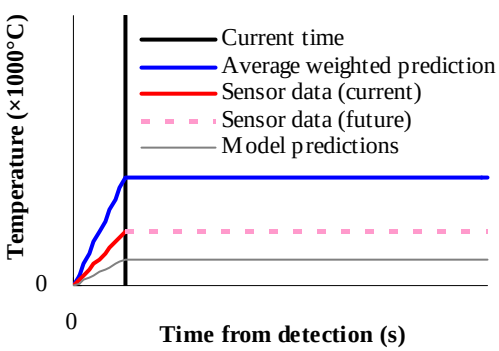
Figures 4.11 to 4.16 compare the hot layer temperature of the sensor measurements and those of the K–CRISP prediction of all four rooms at six different timestamps with legend indicated in graph (b). The sensor measurements are only "known" up to the current instant in time, as indicated by the thick red line, with the dotted portion showing the future trends. The sets of narrower light lines represent a random sample of 100 scenarios (predictions) selected from amongst those generated by K–CRISP in the most recent 30 seconds period. Finally, the evolution of the "average of weighted prediction" is depicted by the thick blue line.

After several applications of the updating procedure it is apparent that the model has been steered in a way that the slope of the temperature rise becomes similar to the real fire. At 120 seconds from detection, the model still generates scenarios with the fire origin in *Room 2*, *Room 3*, and the *Corridor* (see Figure 4.11, (c), (d), and (e)) as the PDF of *fire origin* is not yet converged. At 540 seconds, K–CRISP focuses on those fires in *Room 1* and gradually adjusts the slope of the temperature rise close to the sensor observations. At 810 seconds from detection, the model was able to forecast a "flashover" (when the temperature exceeds 500°C) around 1100 seconds from detection (see Figure 4.13); however, once the model had observed a continuous temperature decrease at around 1100 seconds, it had changed its fire phase and fixed parameters for the *growing* phase, i.e. they could no longer be modified, while releasing those for the *non–growing* phase (see Figure 4.14). In the experiment, the

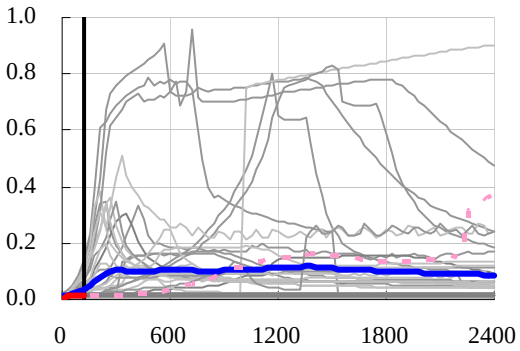
temperature continued to decrease until 2100 seconds from detection, as flashover began to develop (see Figure 4.15). K–CRISP then changed its fire phase from *non–growing* to *full–room* at around 2200 seconds, shortly after the temperature had begun to rise significantly (see Figure 4.16). Although the average temperature in *Room 1* was slightly over–predicted and others were under–predicted during most of the update procedure, K–CRISP generated surprisingly accurate predictions throughout the fire overall.



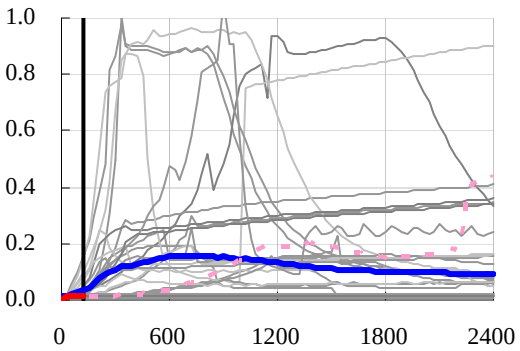
(a) *Room 1*



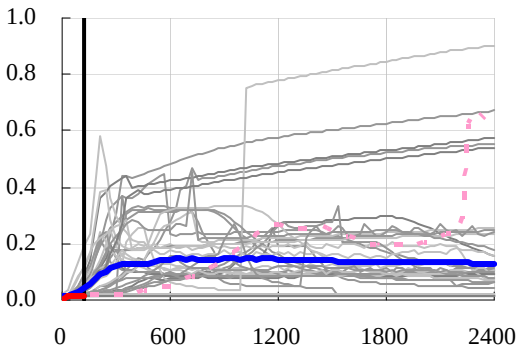
(b) Legend for figure (c), (d), and (e)



(c) *Room 2*

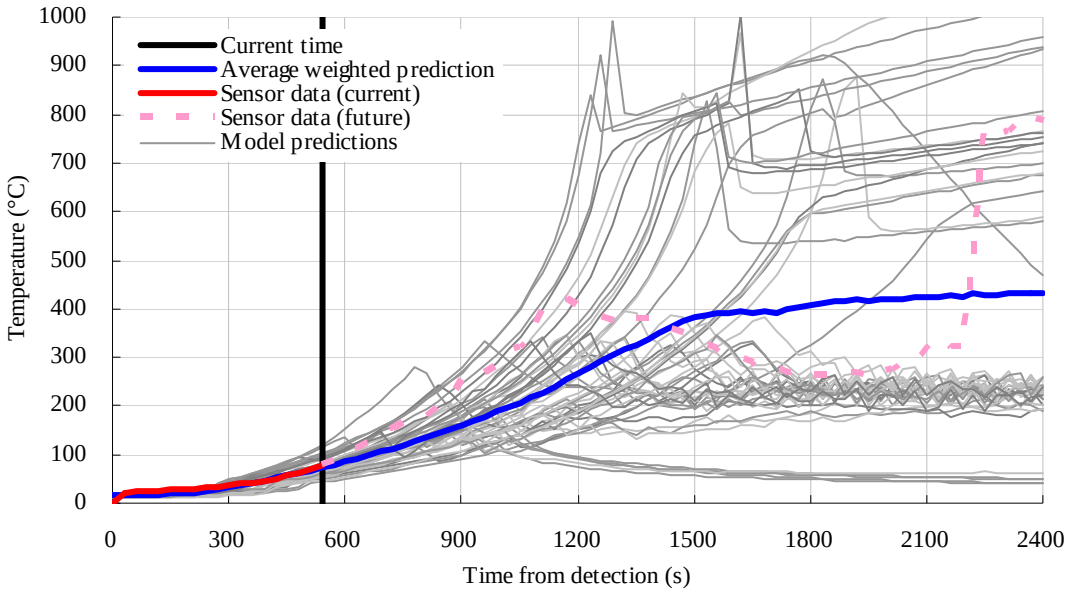


(d) *Room 3*

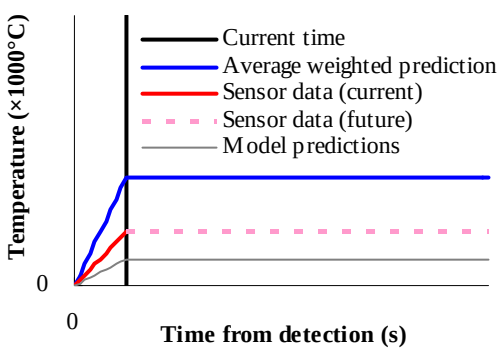


(e) *Corridor*

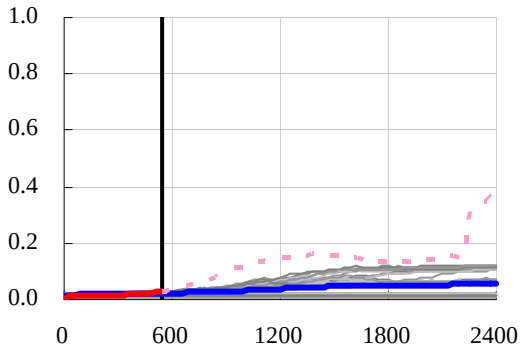
Figure 4.11 – K–CRISP predictions of average compartment temperatures in all rooms (120 s)



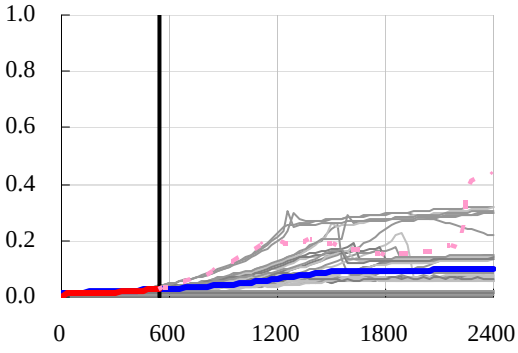
(a) *Room 1*



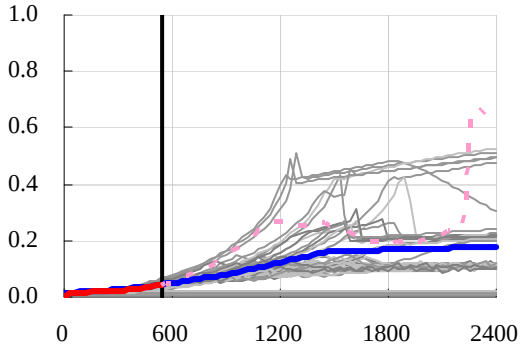
(b) Legend for figure (c), (d), and (e)



(c) *Room 2*

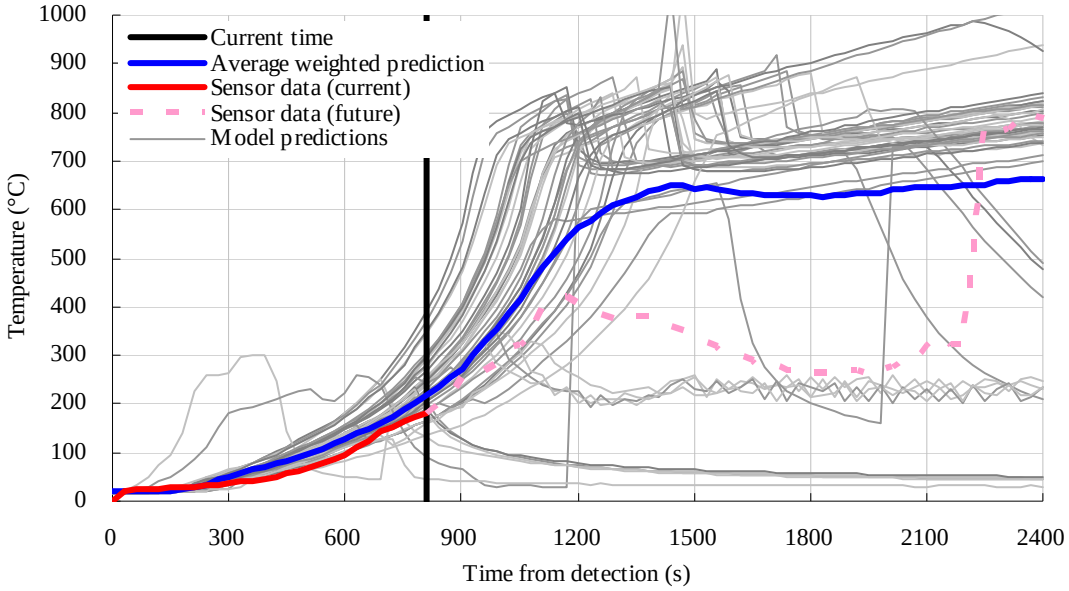


(d) *Room 3*

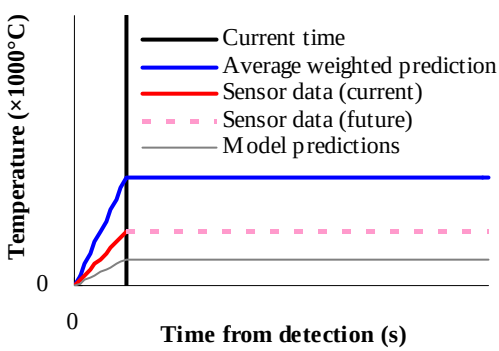


(e) *Corridor*

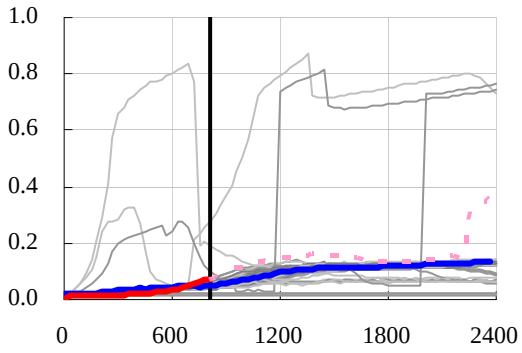
Figure 4.12 – K–CRISP predictions of average compartment temperatures in all rooms (540 s)



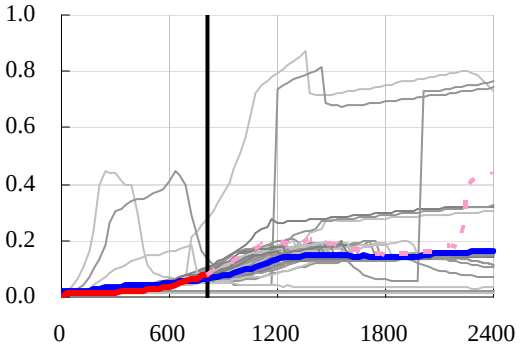
(a) *Room 1*



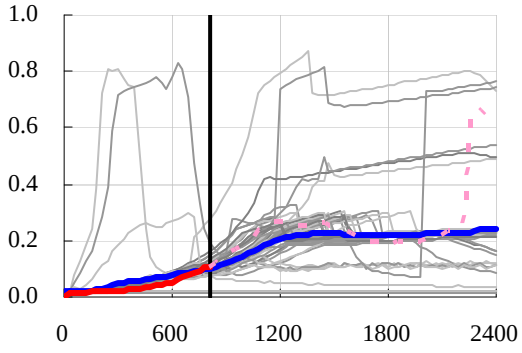
(b) Legend for figure (c), (d), and (e)



(c) *Room 2*

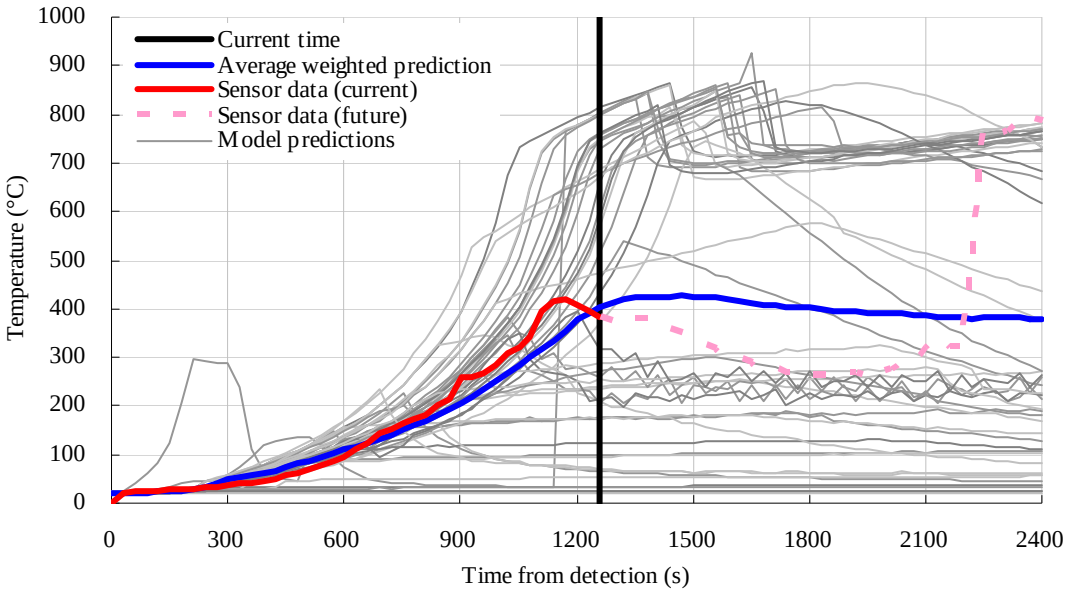


(d) *Room 3*

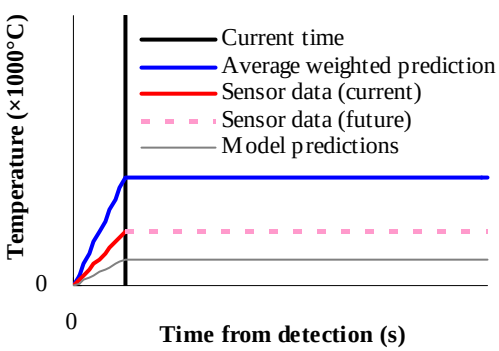


(e) *Corridor*

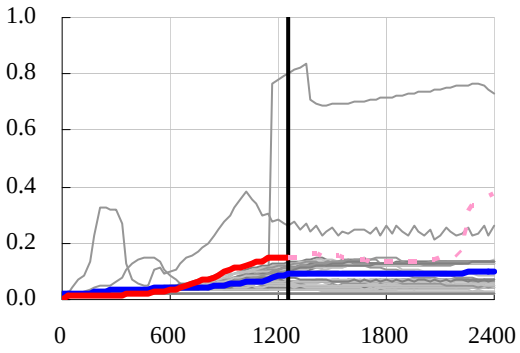
Figure 4.13 – K–CRISP predictions of average compartment temperatures in all rooms (810 s)



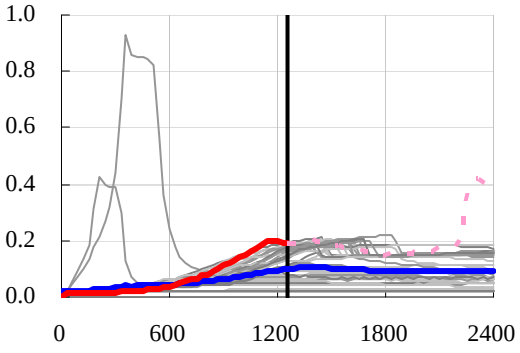
(a) *Room 1*



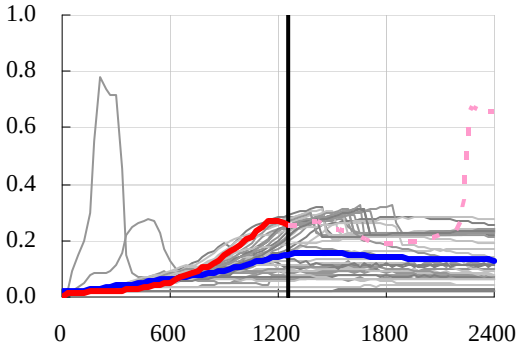
(b) Legend for figure (c), (d), and (e)



(c) *Room 2*

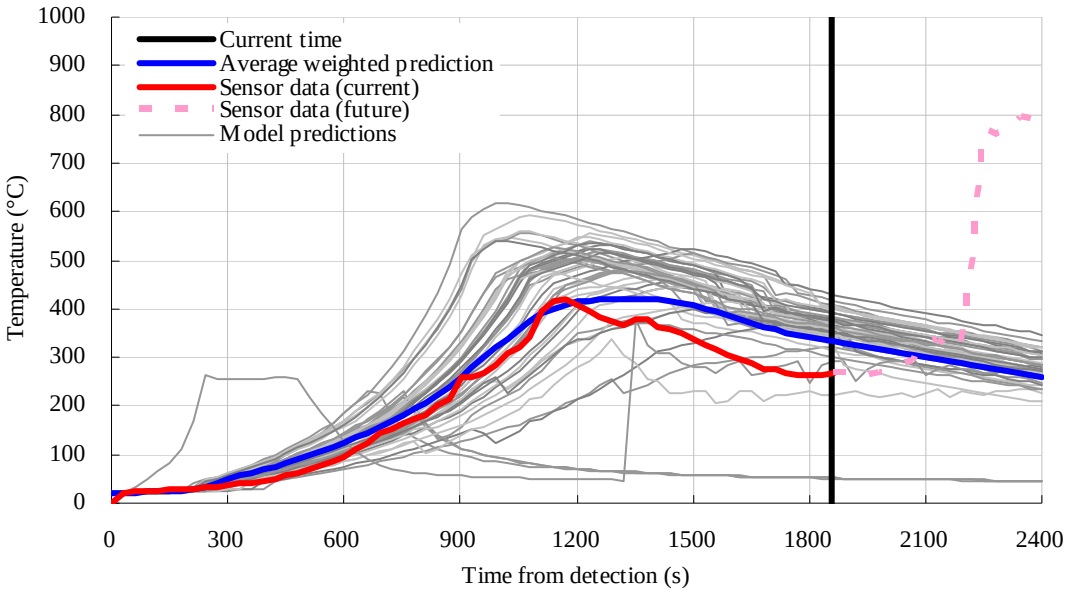


(d) *Room 3*

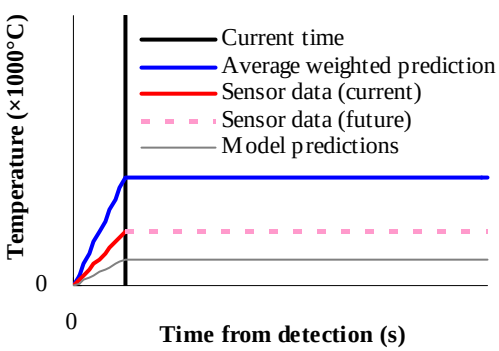


(e) *Corridor*

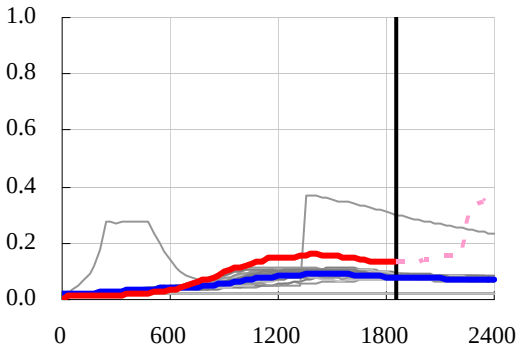
Figure 4.14 – K–CRISP predictions of average compartment temperatures in all rooms (1260 s)



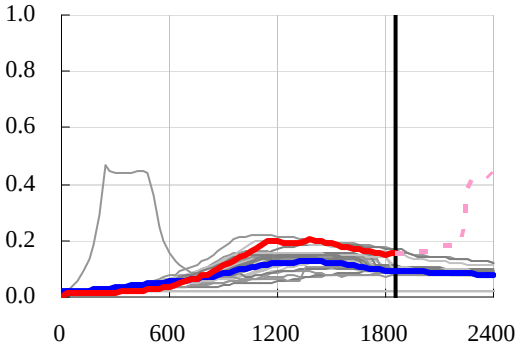
(a) *Room 1*



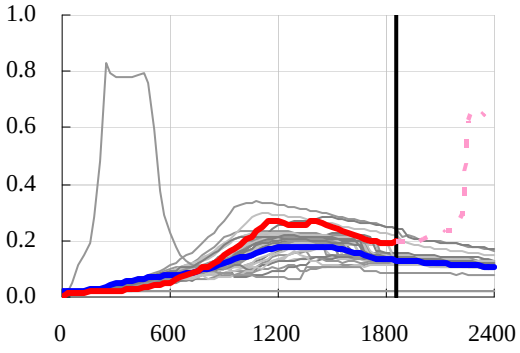
(b) Legend for figure (c), (d), and (e)



(c) *Room 2*

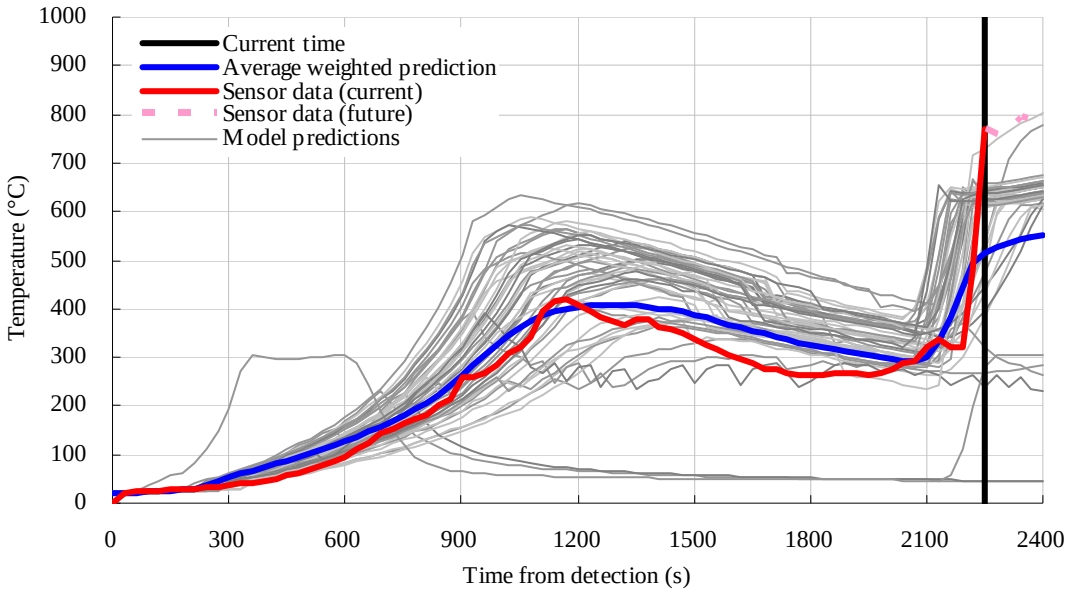


(d) *Room 3*

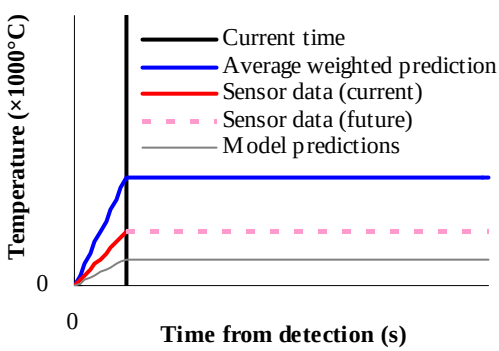


(e) *Corridor*

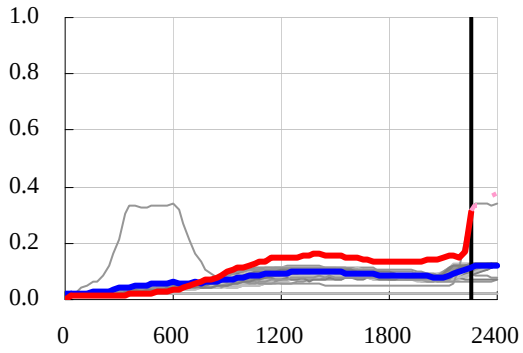
Figure 4.15 – K–CRISP predictions of average compartment temperatures in all rooms (1860 s)



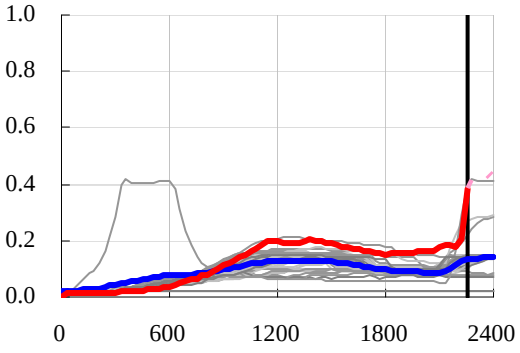
(a) *Room 1*



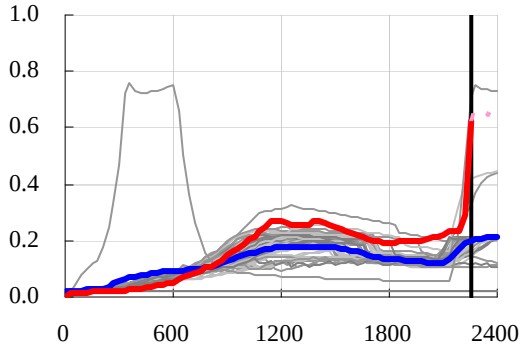
(b) Legend for figure (c), (d), and (e)



(c) *Room 2*



(d) *Room 3*



(e) *Corridor*

Figure 4.16 – K–CRISP predictions of average compartment temperatures in all rooms (2250 s)

Figures 4.17 and 4.18 show examples drawn from one of the predictions which have been selected as they clearly show the difference between “average of weighted predictions” and a simple average of all predictions. As the former parameter contains information about the weighting factor of each scenario, i.e. how close it lies to the sensor observation, it follows the sensor measurement more closely than the simple average in the early stage when the model has not yet been “steered” enough to generate optimum scenarios, as shown in Figure 4.17. But Figure 4.18 shows that even after the model is steered, the weighted average is capable of showing the peak at around 1500 seconds, while the spikes are smoothed in the simple average temperature.

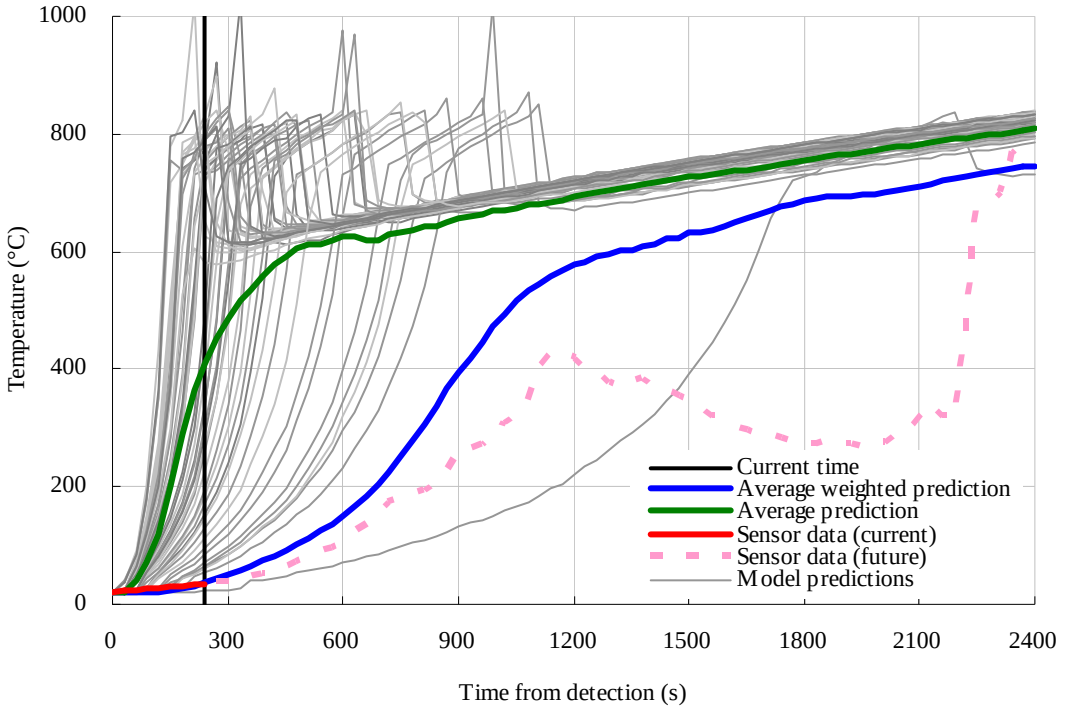


Figure 4.17 – Comparison between “average weighted prediction” and “average prediction” at 240 s

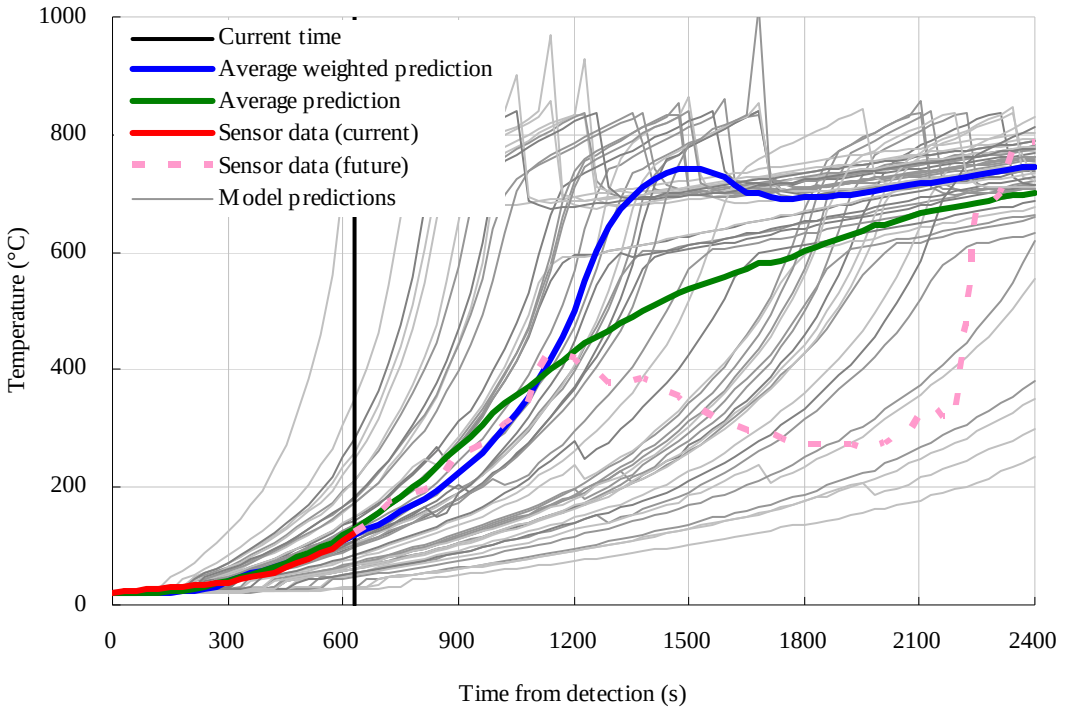


Figure 4.18 – Comparison between "average weighted prediction" and "average prediction" at 630 s

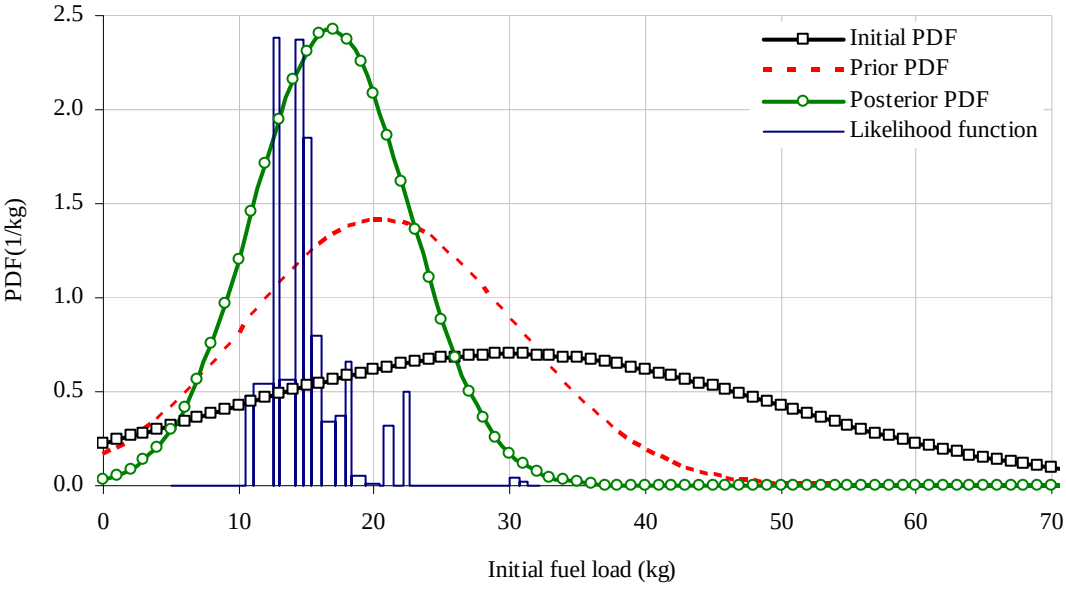


Figure 4.19 – Example of PDF changes after Bayesian inference for *initial fuel load*

Figure 4.19 shows an example drawn from one of the updating procedures which has been selected as it shows the change of the PDF most clearly. The curves represent the current and previous PDFs of the initial fuel load and the columns are a histogram which is the averaged weight of a certain number of scenarios, i.e. the likelihood function. As more sensor observations are supplied to the model, the PDF tends to have a smaller variance; therefore, the model orientates around a specific value of the parameter and eventually the predictions become more focused, being (hopefully) a better match to the sensor measurements.

Figures 4.20 to 4.23 present the convergence of a parameter value in another manner, related to the change of the fire phase defined by the model. In Figure 4.20, the hot layer temperature from the sensors is shown and vertical lines are plotted where the model has determined that the fire has changed its *phase*. The fire phase is first categorised as *initiation*, in which *fire origin* and the *ambient temperature* information are inferred, while all other parameters are fixed. Once the fire is assessed to have progressed into the *growing* phase at around 250 seconds from detection, parameters related to fire growth such as the *rate of flame spread* are fully flexible while the *fuel load*, which does not affect the temperature rise, is fixed to a reasonably large value so that it cannot restrict the fire growth due to early burnout (see Figures 4.21 and 4.22). K–CRISP starts controlling the PDF and the *rate of flame spread* starts converging to a certain value, around 0.001 m/s in this case.

At around 1100 seconds from detection, the fire starts to die down and the model observes several more sets of sensor measurements, eventually deciding to change the phase to the *non–growing*. When the fire starts to decay, parameters such as *rate of flame spread* no longer influence the fire definition. Instead, *fuel load* and *maximum radius of burning surface*, etc., govern the time of the burnout and the slope of the temperature rise; therefore, the PDF of *rate of flame spread* is fixed so that it no longer can be steered and the *fuel load* is released during this phase (see Figure 4.22).

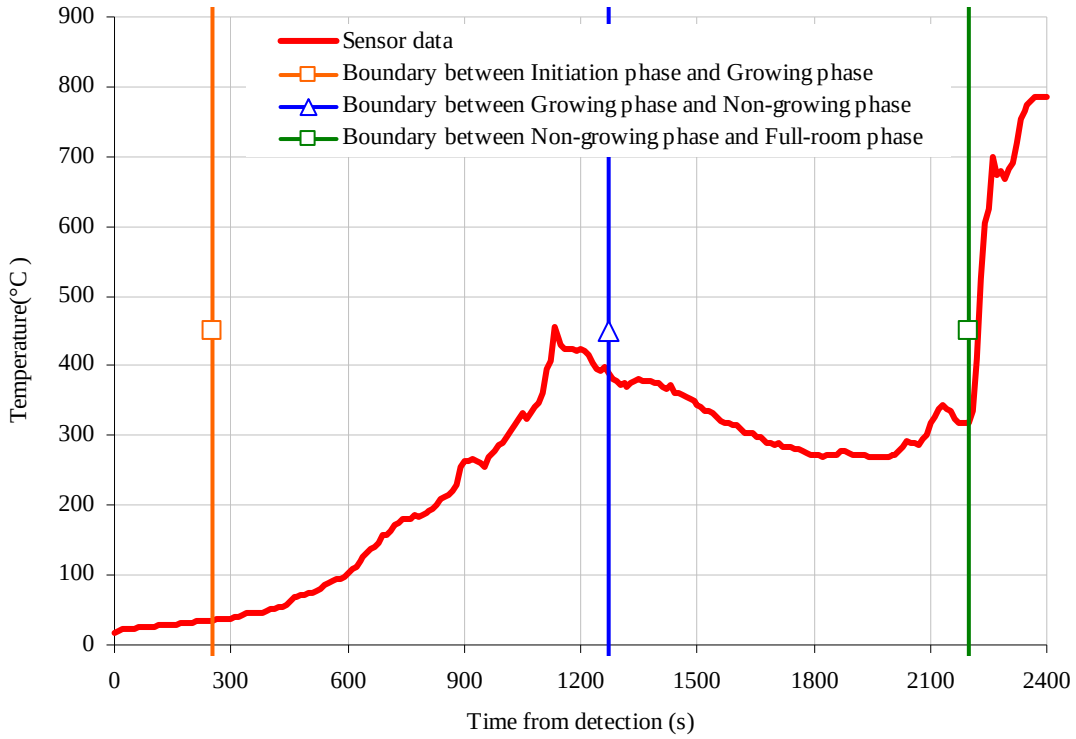


Figure 4.20 – Gas temperature and fire phase definition of K–CRISP

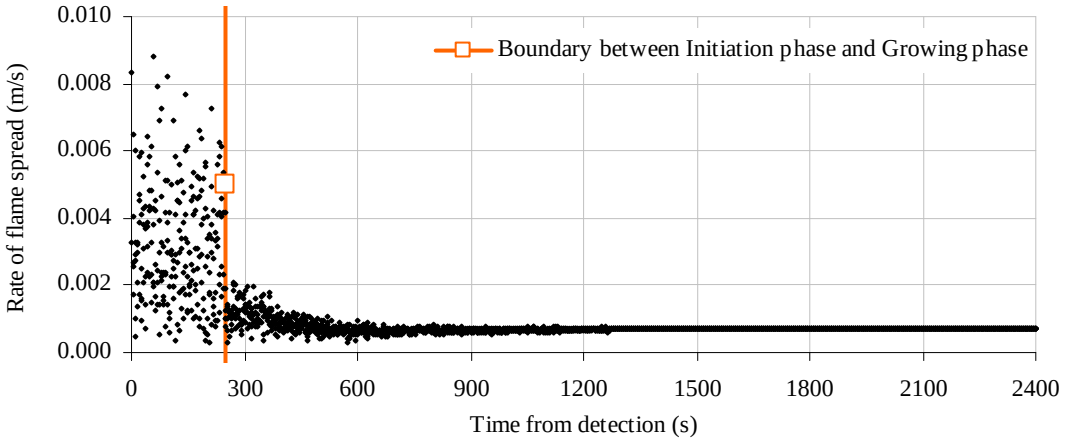


Figure 4.21 – PDF convergence of *rate of flame spread* in K–CRISP simulation

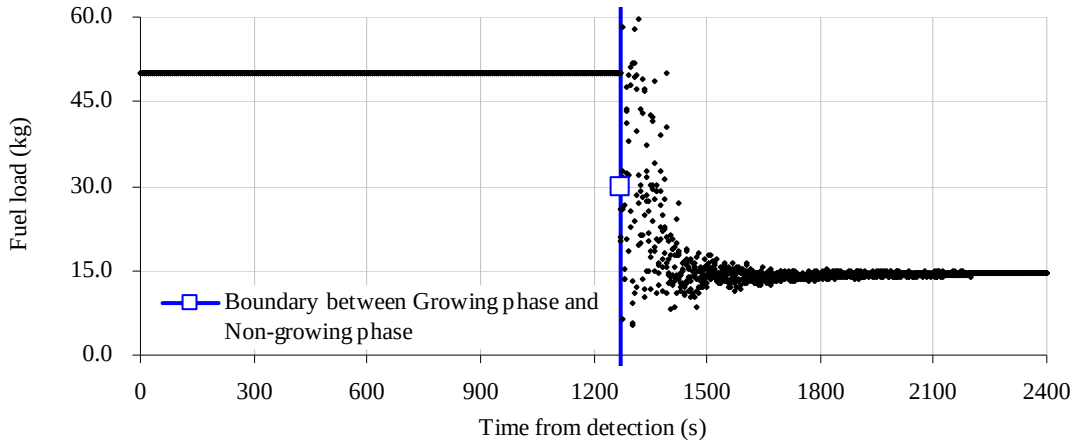


Figure 4.22 – PDF convergence of *fuel load* in K–CRISP simulation

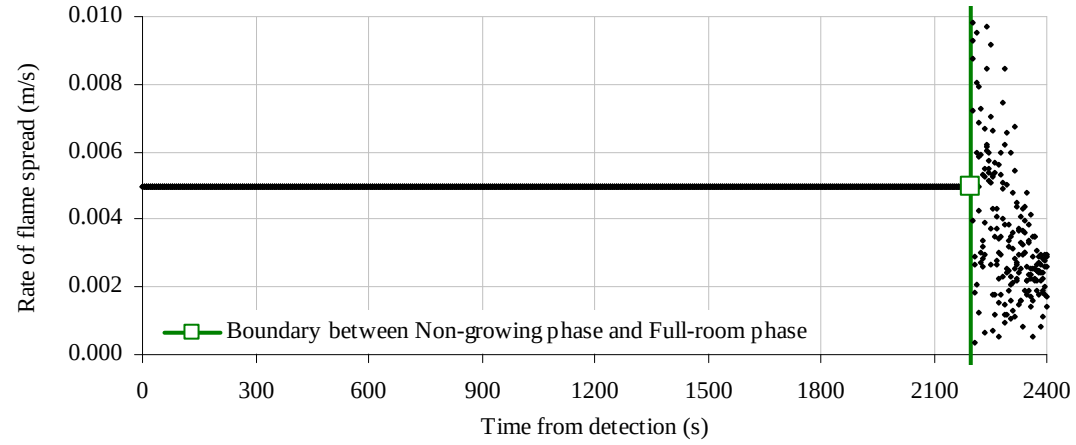


Figure 4.23 – PDF convergence of *rate of flame spread for full–room* in K–CRISP simulation

Around 2200 seconds from detection, the fire starts to engulf the entire room and parameters in the *full–room* category assume the main role of controlling the fire. In practice, the temperature rise was too rapid for the model at this point, so it did not have an opportunity to infer the value of the *rate of flame spread* for the "full–room" item. However, Figure 4.23 shows that the model did try to steer the parameters in the last few minutes before the fire was finally extinguished at about 2850 seconds.

A further exploration of model performance was made using a much larger numbers of scenarios but reverting to a selection via a standard deviation parameter; here, the match was qualitatively inferior to that obtained via the updating procedure described above. This result is expected, because the selection is effectively being undertaken blind, i.e. there is no learning process going on as time progresses. This result alone demonstrates the value of the adopted steering approach in providing a good engineering method for predicting the evolution of the incident. Moreover, there are associated computational issues too, as the current updating approach retains only a modest number of scenarios at any point in time, while procedures based purely on selection with no steering are obliged to select amongst an ever increasing number of scenarios as the incident proceeds. Thus, the burden on the scenario selection computation can rapidly become restrictive at longer durations.

4.4.3 Assessing model predictions of each sensor channel

The results in the previous section showed that the “default” case successfully played a role as initial “sensor observation” and the model was capable of coming up with surprisingly accurate predictions even by using a simple temperature data from each room. Although only four temperature data were involved in the steering procedure in this case, the model did generate “predictions” for all other sensor channels as well. As K–CRISP can simulate various types of sensor data such as gas velocity, species concentration, and structural integrity, it begs the question whether these data can also match with sensor observations while they were not involved in the steering procedure.

Figures 4.24 and 4.25 present gas temperatures in *Room 1*. The red line indicates the value measured by a selected thermocouple sensor and the grey lines represent the K–CRISP predictions of the corresponding sensor. The concept of “average weighted prediction” was applied only to the model predictions such as temperature in each room, hot layer depth, the number of people and their FED, and structural deflections and structural integrity, which

was relevant to the ‘Agent–Based Command–Control Component’ in the FireGrid project where the final prediction from K–CRISP is interpreted and converted to more intuitive format. The model results which were not included in the ‘final’ prediction could also be averaged with the weight information in principle, though here a simple average of the predictions was used instead for expediency, indicated by the green line.

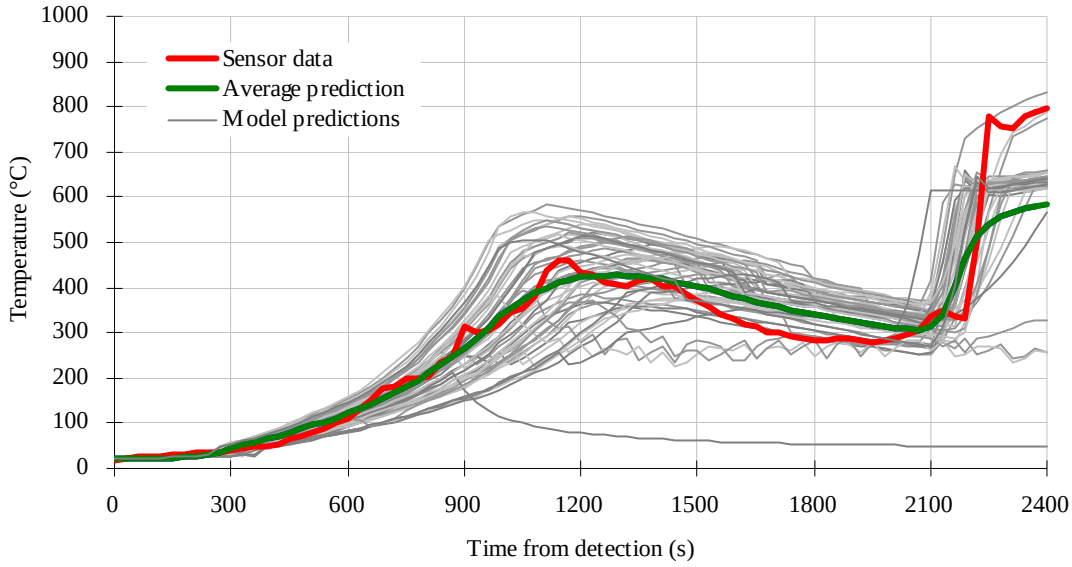


Figure 4.24 – Gas temperature (*Room 1*, channel 7, 1500 mm height)

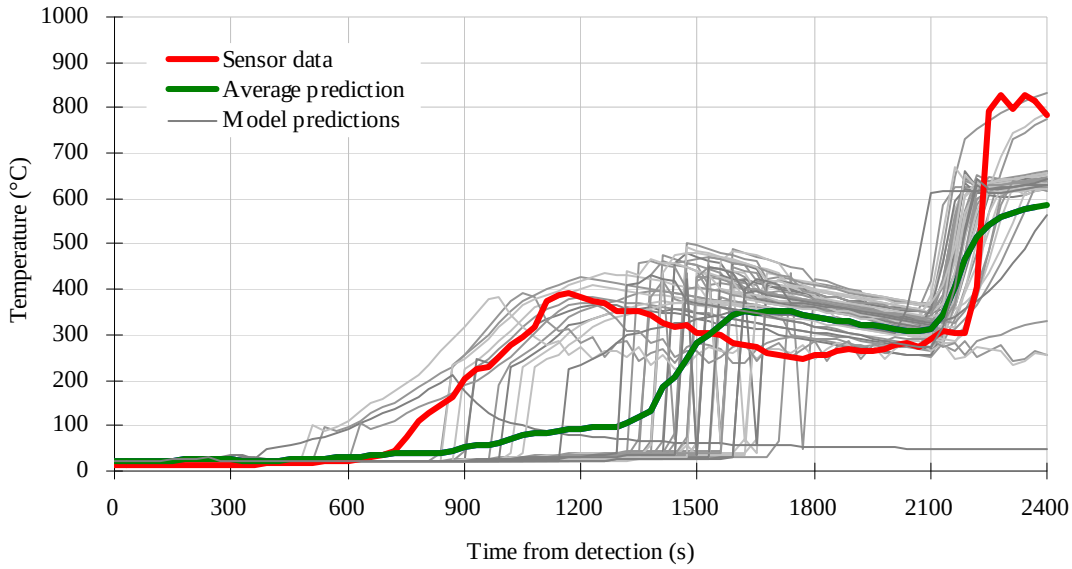


Figure 4.25 – Gas temperature (*Room 1*, channel 10, 600 mm height)

While predictions match quite well with sensor observations when the thermocouple is installed relatively high, thus, giving the hot layer temperature as shown in Figure 4.24 and the figure of the “overall predictions” in the previous section, there is a significant jump in the temperature when the sensor is installed close to the floor (see Figure 4.25). This is due to the fact that zone model has only two layers in each room, i.e. hot layer and cold layer, therefore, the cold layer temperature is suddenly replaced with hot layer temperature when the layer interface crosses over the point where the sensor is installed. This significant rise in the temperature can potentially influence the goodness–of–fit test. For this reason, the average temperature of each room was used in the steering procedure, as shown in the previous section, considering the difficulty of measuring the hot layer temperature in a realistic building.

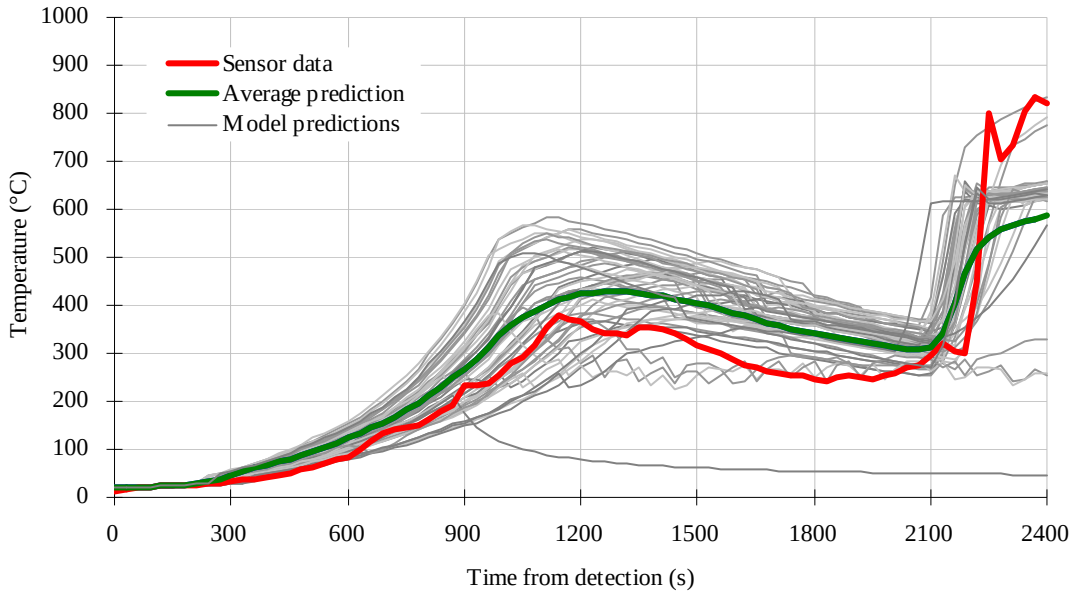


Figure 4.26 – Vent–flow temperature (*Room 1 – Corridor*, channel 13, 1800 mm height)

Figures 4.26 and 4.27 show vent–flow temperatures measured at the door between *Room 1* and the *Corridor*. As the vent–flow temperatures are governed by the gas layer temperature of both rooms, both figures show similar temperature distributions as the room temperature.

However, they share the inherent limitation of having significant rises when the sensor is installed relatively low, closer to the floor. This sudden change in the temperatures is observed in most of the gas temperature and vent–flow temperature values in other rooms.

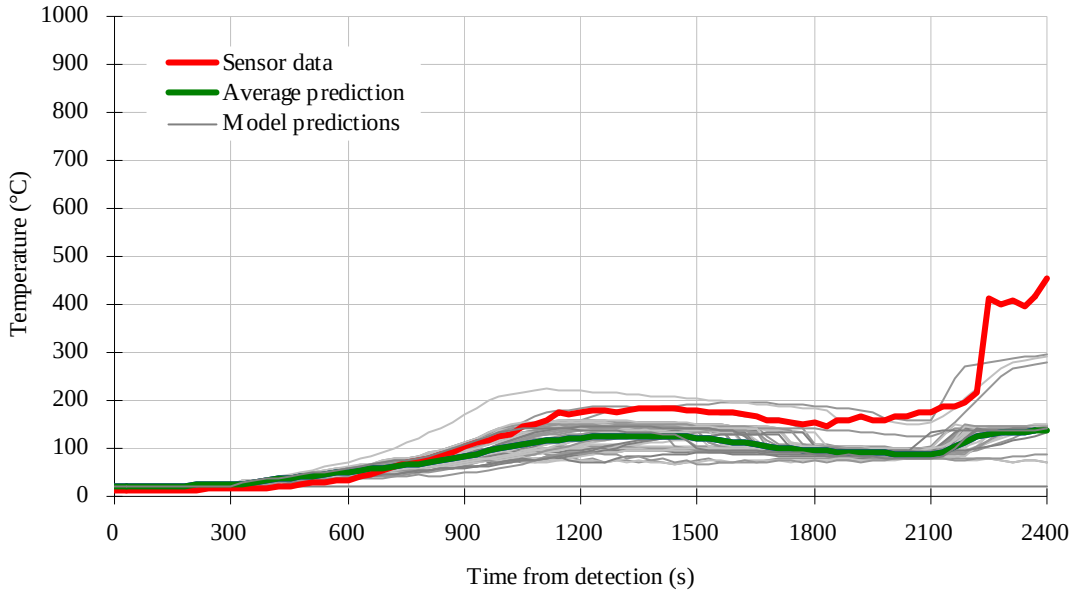


Figure 4.27 – Vent–flow temperature (*Room 3* – outside, channel 91 1900 mm)

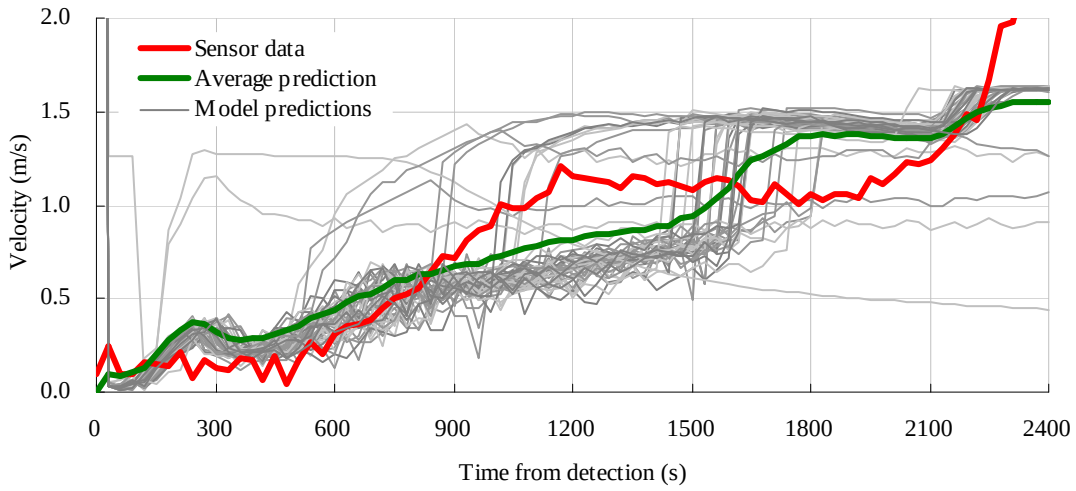


Figure 4.28 – Velocity (*Room 1 – Corridor*, channel 64, 100 mm height)

The velocity of the vent–flow was also measured by the velocity probes at the doorways between *Room 1* and the *Corridor*, and between *Room 3* and outside. Figure 4.28 is one

selected vent-flow velocity measurement among 9 sensors. Although the model prediction was slightly over–predicted here and there, the prediction could still be described as "reasonable" since it shows the overall trend of the velocity change fairly well.

As mentioned earlier, K–CRISP can generate concentration rates for O_2 and combustion products, i.e. CO_2. Figures 4.29 and 4.30 indicate O_2 and CO_2 gas concentration in the *Corridor* and the predictions gave a good fit to the sensor measurements throughout the fire development.

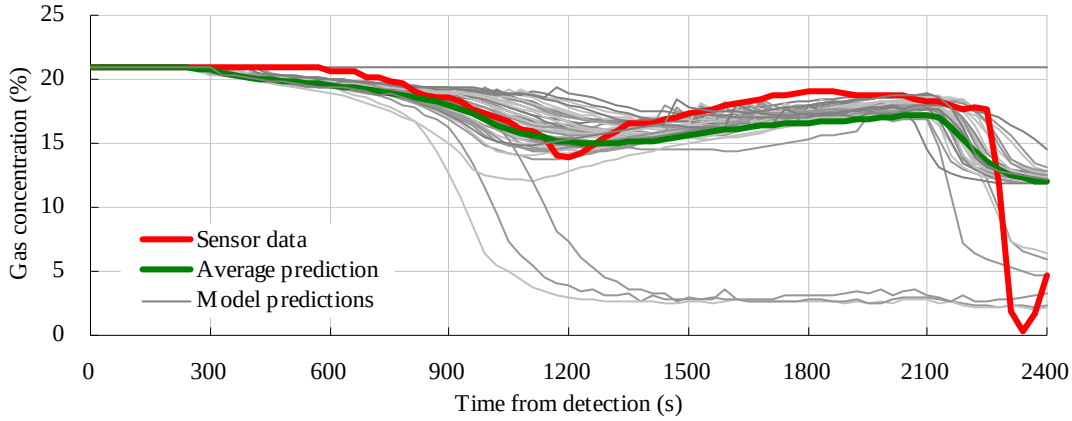


Figure 4.29 – O_2 concentration (*Corridor*, channel 56, 1500 mm height)

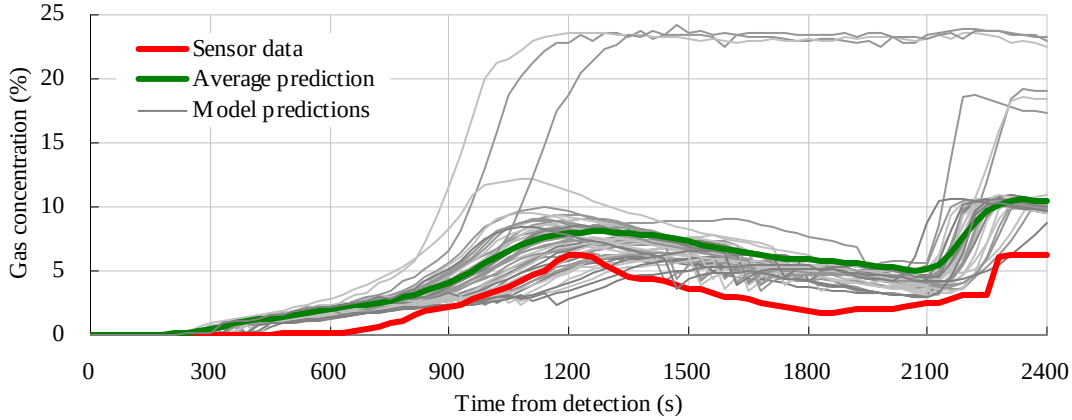


Figure 4.30 – CO_2 concentration (*Corridor*, channel 58, 1500 mm height)

During the FireGrid D7.4 demonstrator, wall surface temperatures were measured, both inside and outside of the test rig. All the wall surface temperatures measured outside of the rig and those of the model predictions showed similar values to the ambient temperature as they were not directly affected by the fire or smoke. But some sensor measurements showed a temperature rise when the walls were partially damaged in the *full–room* phase. Those measured inside the test rig, on the other hand, are interesting because the temperatures predicted by the model were under–predicted when the sensor is installed closer to the fire while they show a good match to the sensor data when it is installed further from the fire (see Figures 4.31 and 4.32). This is probably because the zone model in K–CRISP currently does not have a capability of representing the orientation of individual burning items; therefore, the fire impacts each wall surface with the same amount of heat flux.

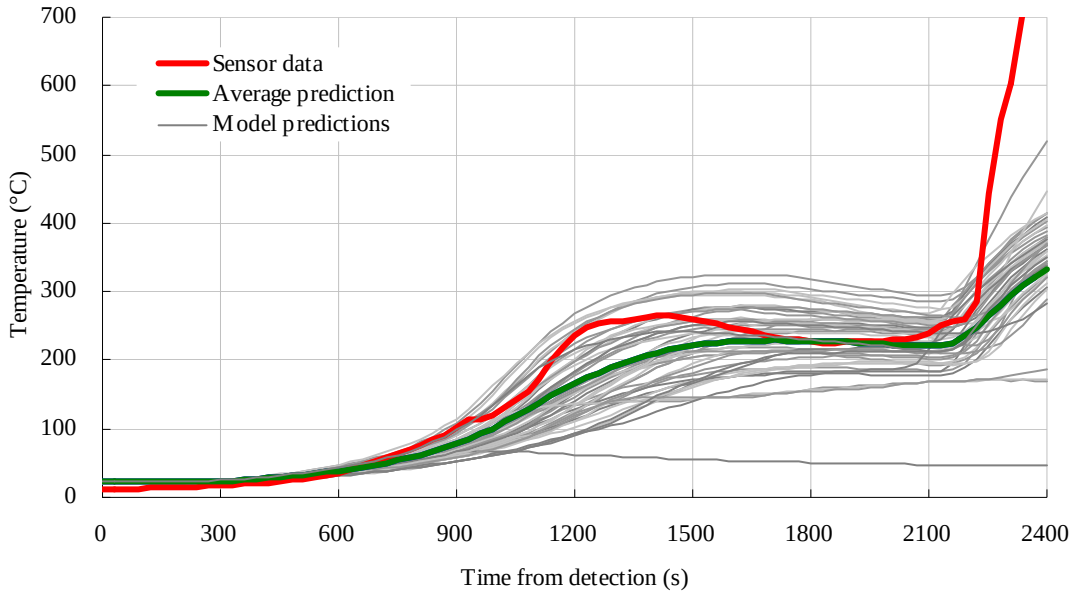


Figure 4.31 – Wall temperature (*Room 1*, channel 69, far from fire source)

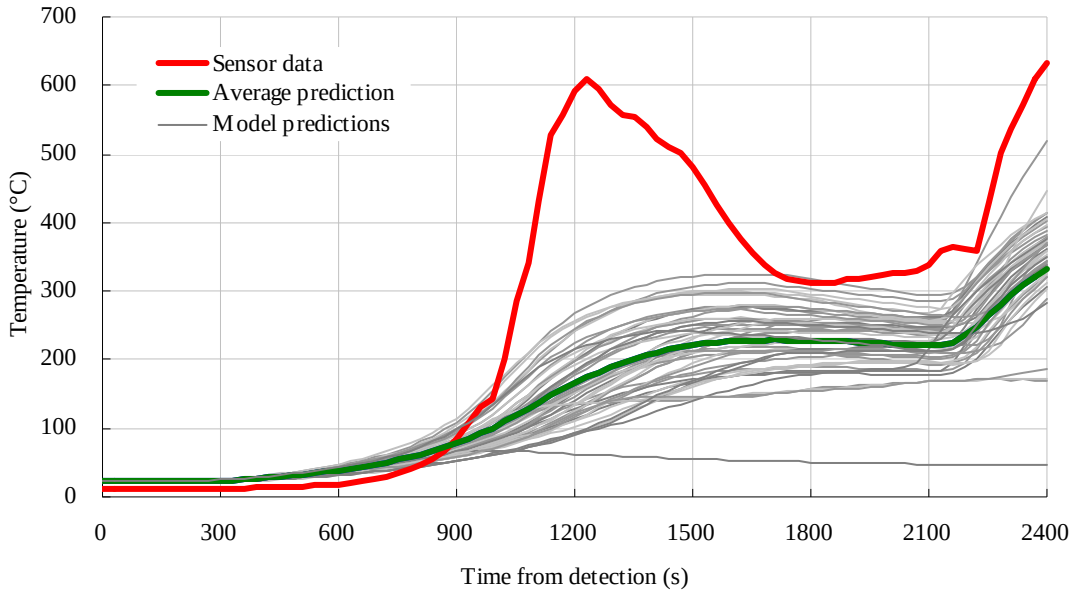


Figure 4.32 – Wall temperature (*Room1*, channel 67, close to fire source)

Besides the fire status information the model was also generating predictions of structural temperature and structural integrity. During the test these correctly indicated probable failure of a structural truss placed within the room of fire origin, as actually occurred towards the end of the fire. Again, more careful analysis was undertaken later using a replay of the sensor data. Figure 4.33 shows the temperature of the truss measured by one selected sensor among 9 thermocouples installed on the structure. The predicted temperatures match quite well with the sensor measurement of this channel as well as most of the other sensors. Figures 4.34 and 4.35 compare the evolution of the experimental measurement and predictions of the vertical deflection at the centre of the truss, with the predictions being a weighted average derived from the range of fire curves generated at any particular instant. For this member the transition from collapse being “possible” to being “probable” occurs at a mid–span deflection of 0.25 mm. Figure 4.34 shows the prediction early in the fire at a time of 300 seconds from detection. Even at this stage the greatly increased likelihood of collapse at times approaching 1100 seconds is indicated, with the transition from possible to probable occurring at around 1070 seconds; also shown is the actual measured deflection –

this is initially slightly negative (that is, upward) presumably due to the fire exposure of the lower surfaces of the member, a factor not considered in the simple model; nevertheless, the actual behaviour was comparable to the prediction, with a deflection of 0.25 mm reached at 750 seconds. The failure values are also in accordance with the fire temperature rising toward 500°C, beyond which steel begins to rapidly lose strength [43].

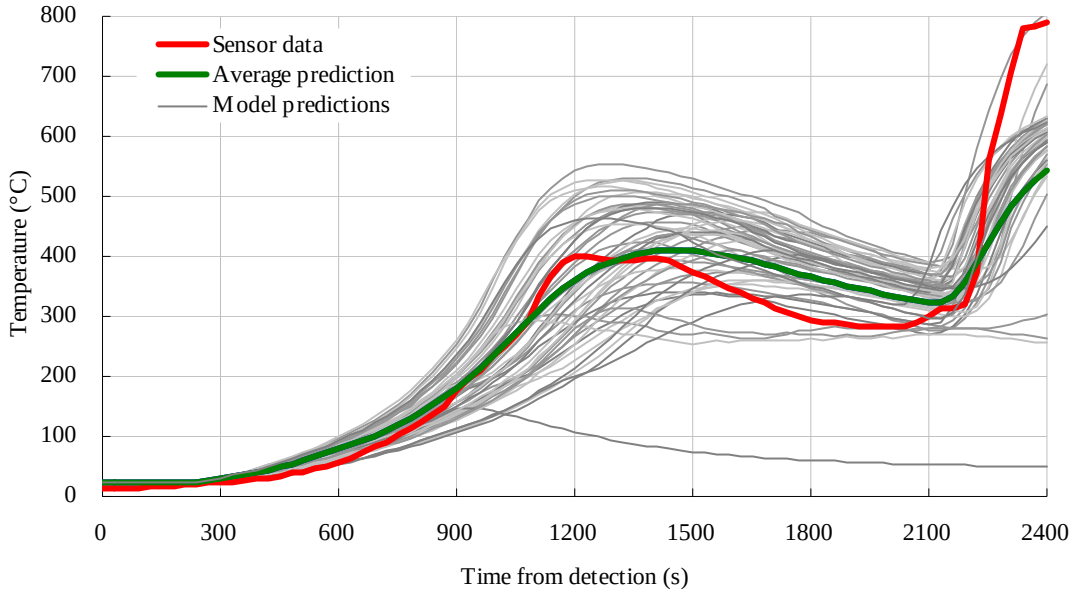


Figure 4.33 – Structural temperature (Truss, channel 100)

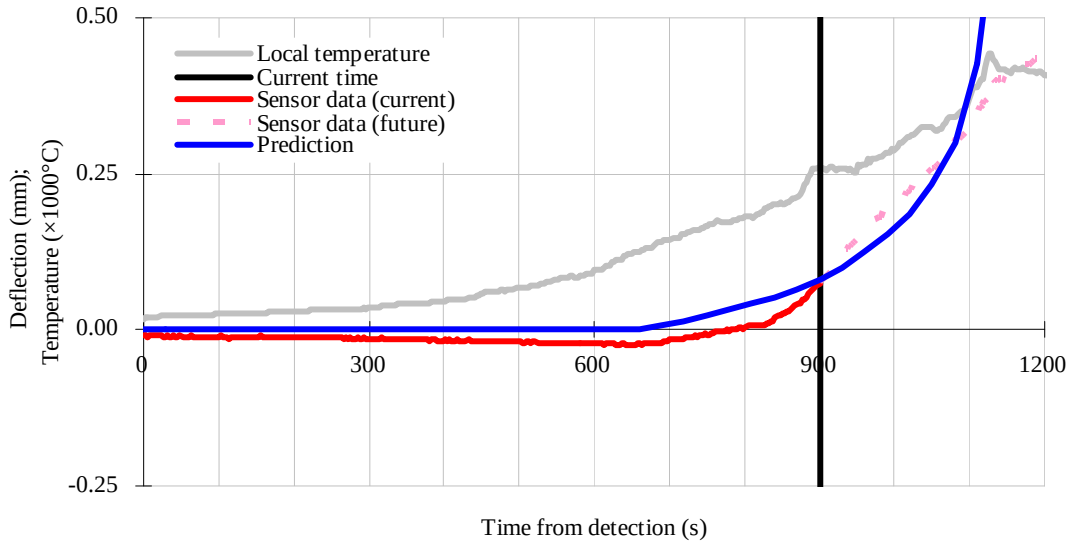


Figure 4.34 – Comparison between sensor measurement and predicted structural deflections at 300 s

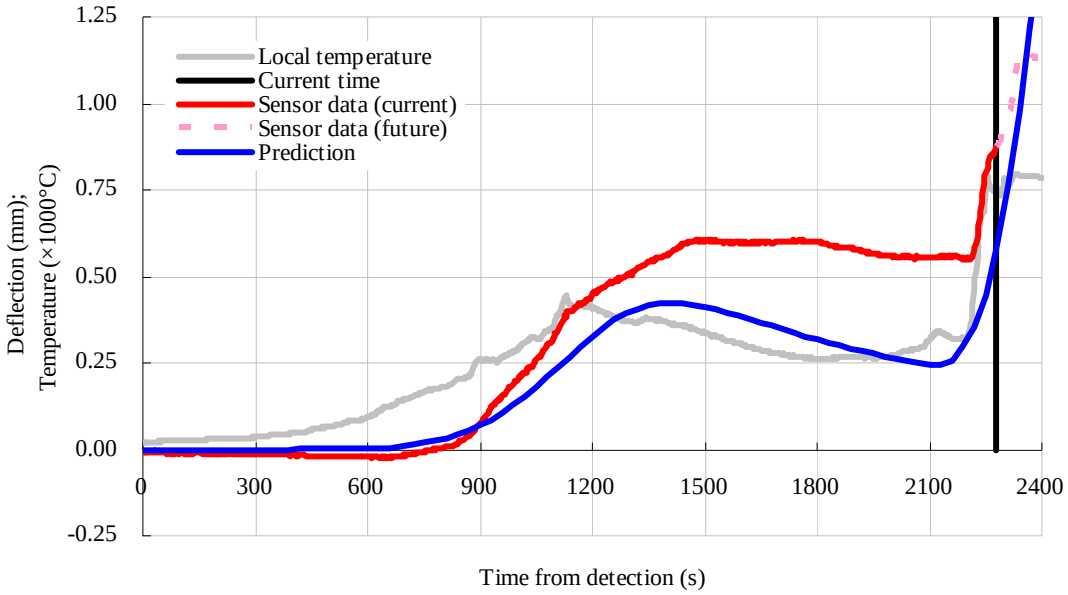


Figure 4.35 – Comparison between sensor measurement and predicted structural deflection at 2100 s

After the fire behaviour changed at around 1100 seconds, the predicted structural performance was revised (see Figure 4.35), with some recovery of the member up to a time of about 2100 seconds, a trend also mirrored in the deflection measurements; however, following flashover at around 2200 seconds a rapid further deterioration ensued, induced by the sudden increase in fire exposures. Total collapse of the truss followed at around 3300 seconds during the cooling phase of the fire, extinguishing water having been applied at about 2700 seconds.

The model successfully reproduces the structural temperature and its collapse, even when the lookup for the member temperature is derived from the thermal model in K–CRISP rather than from direct measurements. As the model was capable of predicting the temperature of the truss, it is in principle able to steer the model using structural temperatures. Additionally, if the structural temperature itself, which is directly measured from the truss, was used in the SIP analysis rather than the predicted values, the prediction can even be improved. It should, however, be noted that it is still extremely challenging to predict the exact point in time of

the structural failure, as there are many other uncertain factors that affect the structural collapse.

Finally, human behaviour aspects are not specifically considered here, though inferences related to them were automatically being derived by the model in the course of the fire simulations. While these were largely redundant for an unpopulated scenario, i.e. the model is only making inferences about the lack of human activity in the building, for any case where such activities do affect fire behaviour and development the model can in principle deduce what has occurred, e.g. a door opened or closed, and thereby some deductions about human behaviour and egress. An example of this type of inference, based on a hypothetical care home scenario, is described elsewhere by Fraser–Mitchell *et al.* [44].

The results in this section show that K–CRISP is capable of generating predictions using a simple characteristic input, i.e. average gas temperature of each room. It is a little surprising, but encouraging that the model was able to match not only the average temperatures but also those data which were not involved in the steering procedure, such as vent–flow temperature, O_2 and CO_2 concentration, and structural deflection. However, it should be noted that the estimation of other parameters such as oxygen concentration is beyond the model's capability, thus more sophisticated tools (e.g. CFD) may be require to truly predict those parameters.

4.4.4 Assessing model prediction with various initial conditions

In the previous section, K–CRISP was tested in the environment where fairly abundant information, i.e. average temperatures of all rooms, is given. In the realistic fire case, however, sufficient information may rarely be supplied and the fire development may not be clear as it had been in the Dalmarnock fire tests and the FireGrid D7.4 demonstrator. Not all rooms in a building may have temperature sensors and fire may start in a room where no

sensors are installed. In such cases, the model has to deduce where the fire started only using sensor data from adjacent rooms. In this section, K–CRISP was run with some level of "lack of information" and the prediction results were assessed.

Prediction without sensors in room of fire origin

Firstly, it was assumed that the model does not have any information from the room of fire origin; instead average temperatures from all other rooms were supplied. Figure 4.36 shows the change of the PDF for *fire origin* parameter. Even without any information from the fire origin, e.g. *Room 1*, the model was cable of inferring that the fire started in *Room 1* after several steering procedures. This was because there was no significant temperature rise in the other rooms during the early stage of the fire development, thus giving bigger weights to those scenarios which had a fire in *Room 1*.

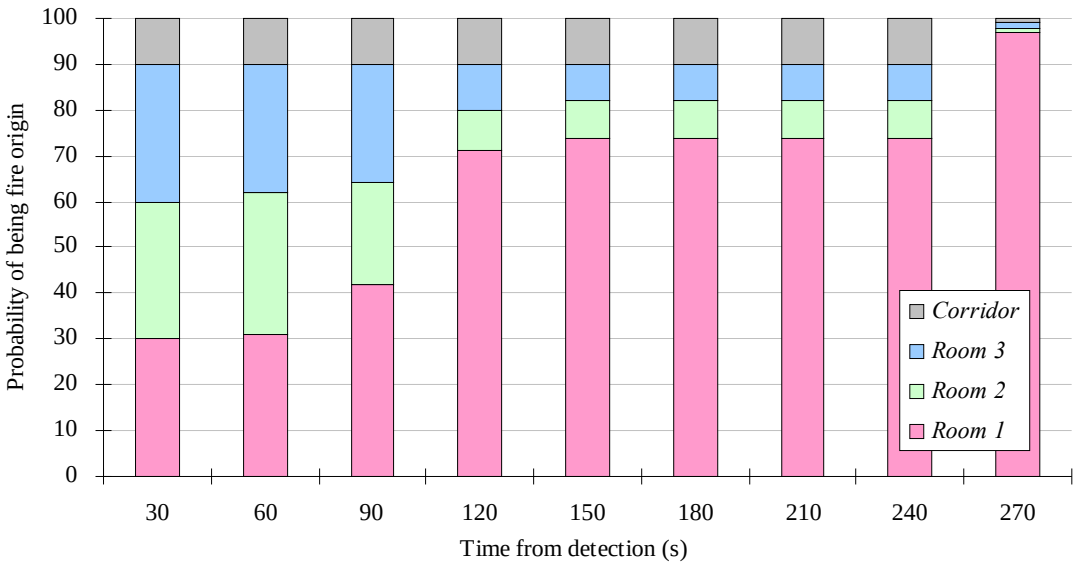


Figure 4.36 – Change of PDF for *fire origin* when no sensor data from *Room 1* is used

Figure 4.37 shows the change of the PDF of the *door status* parameter which indicates the chance of each door being open. As indicated in the previous section, the PDF for *door status* was fixed until the fire origin is inferred at about 300 seconds from detection. In the

earlier test when the average temperatures from all rooms were used, K–CRISP deduced the status of the door between *Room 1* and the *Corridor* (see Figure 4.10) first as the temperature in *Room 1* rose most rapidly. When no information for the fire origin is given, however, the model initially 'thinks' the door is closed and it does not change the PDF until about 540 seconds from detection. This is because K–CRISP assumes the door is open only when smoke comes out from the room but the temperatures of *Room 2*, *Room 3*, and the *Corridor* were not high enough during this period. Once those temperatures rose significantly, however, the model was able to steer all the PDFs to 95% at about 900 seconds from detection.

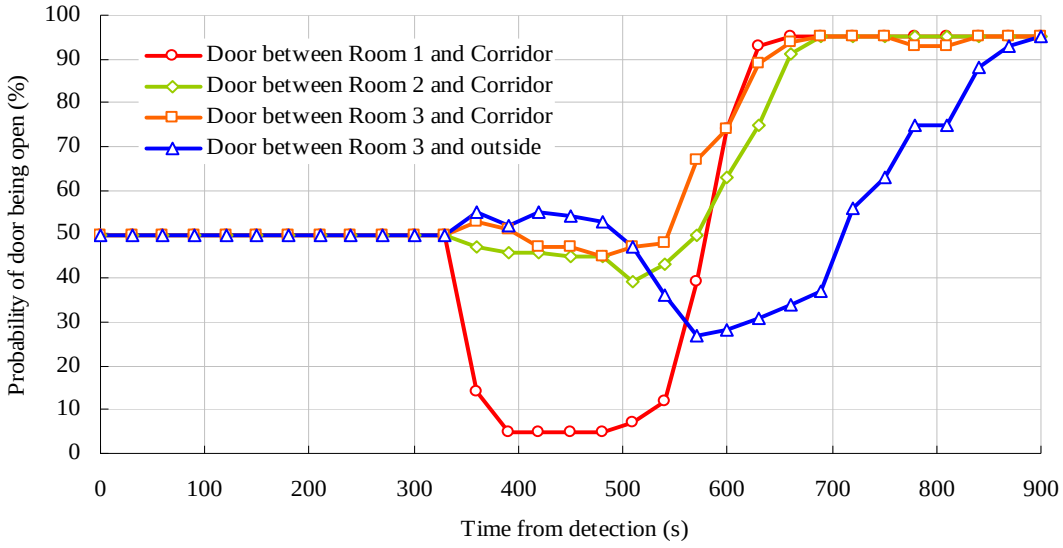
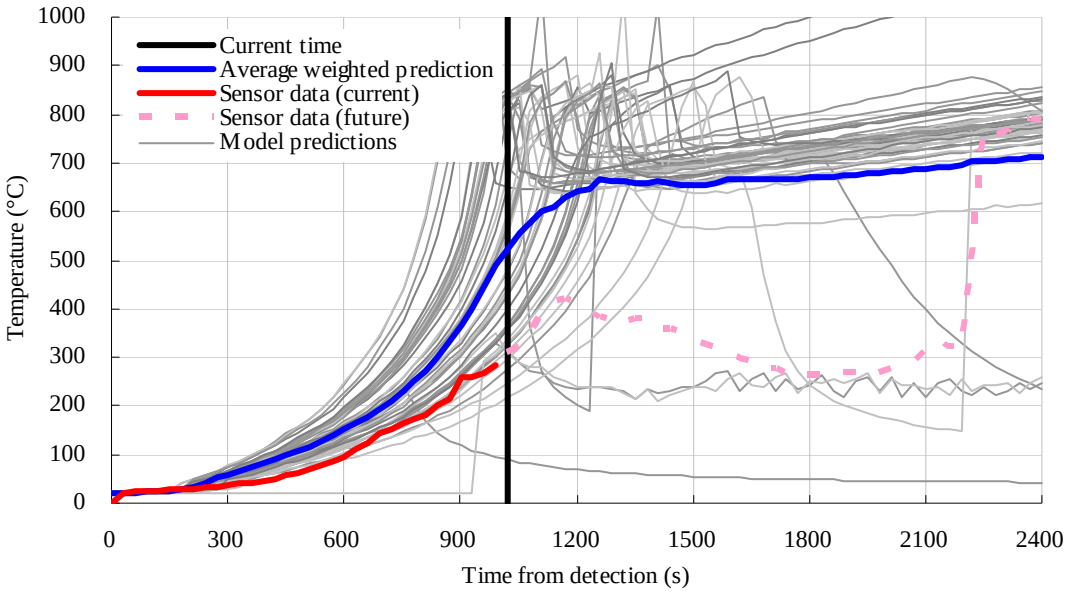


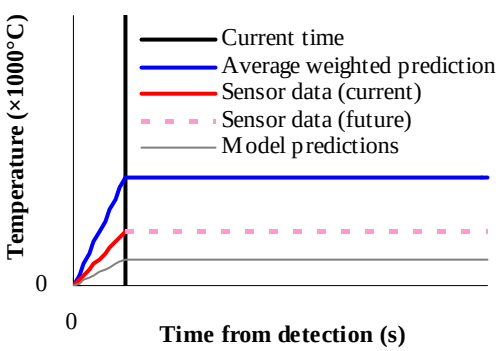
Figure 4.37 – Change of PDF for *door status* when no sensor data from *Room 1* is used

Figures 4.38 and 4.39 compare the hot layer temperature of the sensor measurements and those of the K–CRISP prediction of all four rooms at two different timestamps of the same case. As per the figures in the previous section, the legend is indicated in graph (b) and the sets of narrower light lines represent model predictions. The blue line indicates the average of weighted prediction. Until about 540 seconds from detection, there was no significant difference in the model prediction with those of the 'default' case; however, the prediction of

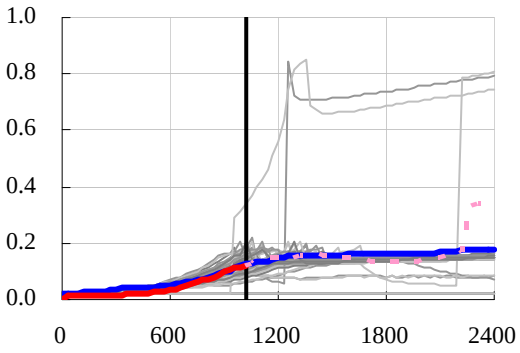
the average temperature in *Room 1* starts to diverge from the sensor observation and is over–predicted by about 200°C at 1020 seconds, as shown in Figures 4.38 and 4.39. This is because K–CRISP did not take the *Room 1* temperature into account for the steering procedure; instead the model focuses on those scenarios which have a better match in terms of the temperature of *Room 2*, *Room 3*, and the *Corridor*. Moreover, as no temperature data is derived from the fire origin and the temperatures from the other rooms does not show the fire development clearly, the model was not able to define the fire phase correctly; therefore, it could not change the phase to the *non–growing* at about 1200 seconds, nor to the *full–room* phase at about 2200 seconds.



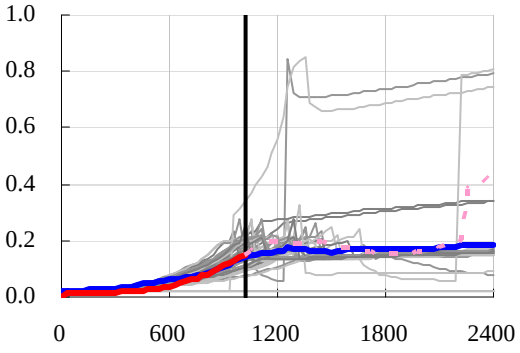
(a) *Room 1*



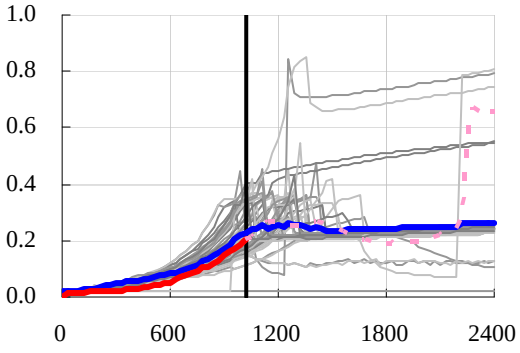
(b) Legend for figure (c), (d), and (e)



(c) *Room 2*

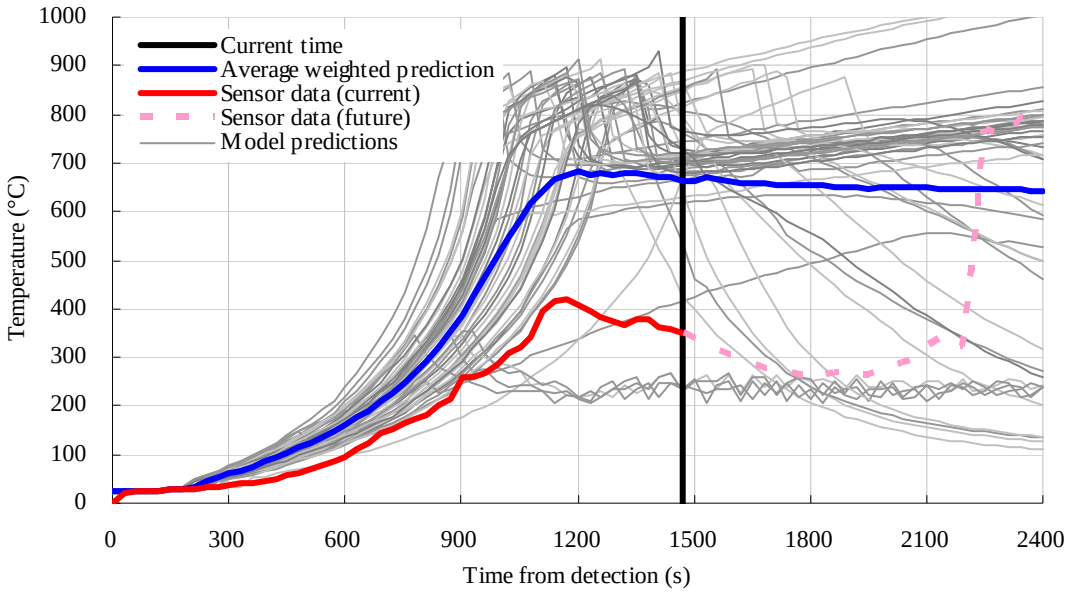


(d) *Room 3*

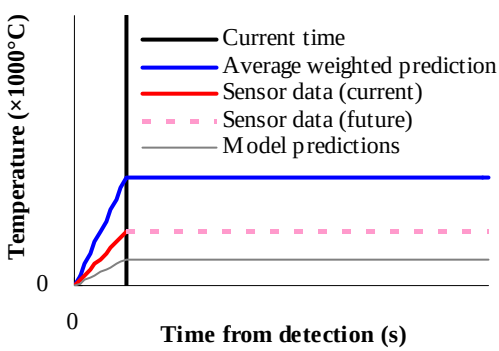


(e) *Corridor*

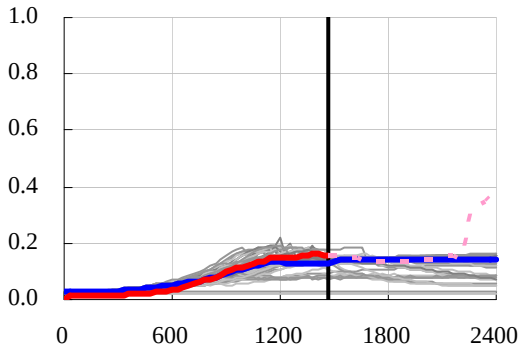
Figure 4.38 – K–CRISP predictions of average compartment temperatures in all rooms (1020 s)



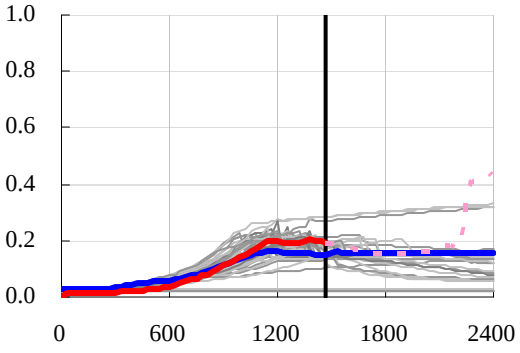
(a) *Room 1*



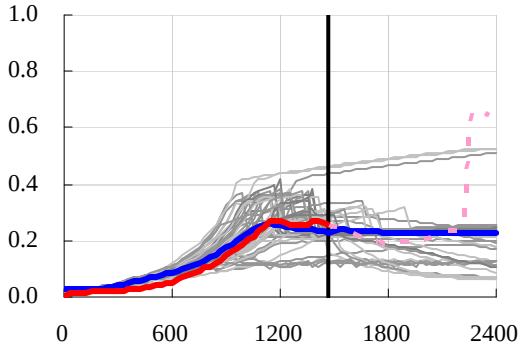
(b) Legend for figure (c), (d), and (e)



(c) *Room 2*



(d) *Room 3*



(e) *Corridor*

Figure 4.39 – K–CRISP predictions of average compartment temperatures in all rooms (1470 s)

The results above show that K–CRISP was still able to deduce the fire origin even when no information of *Room 1* is given, if the average temperatures of surrounding rooms are known. However, it was very difficult to define the fire phase correctly as the fire room temperature was not given to the model. It was also indicated that it is extremely challenging to accurately predict the fire development beyond the *growing* phase if the fire phase is not properly defined.

In order to run the model without the fire phase definition, all the input parameters should be fully randomized throughout the fire. While some parameters such as *rate of flame spread* and *latent heat of vaporization for material* have a relatively small range in terms of "reasonable" values, *fuel load* and *secondary ignition time* can have very wide parametric space, while they can directly affect the point in time of fire being in decay or having a flashover. *Secondary ignition time*, in particular, was set based on the continuous observation of the compartment temperatures when the fire *phase* is turned on; however, the model has to solely depend on a probability to ignite the next item if the fire *phase* is absent. For this reason, if the initial PDF for *fuel load* and *secondary ignition* is set close to the actual value by any chance, K–CRISP will still be able to predict the overall trend of the fire development in spite of the over–prediction of the temperature in *Room 1*. But, if on the other hand the PDF is set far from the actual value, the model is likely to struggle to come up with optimal scenarios and it may predict either no flashover at all or too early a flashover in all of its scenarios.

In the case of initial *fuel load* parameter, however, this issue can still be overcome if supplementary information is given. As described in the early chapter, the sofa used in the FireGrid D7.4 demonstrator was the same sofa used in the Dalmarnock fire tests. The heat release rate for the sofa was measured in the laboratory prior to the test (green line in Figure 4.40). Neglecting the initial peak which corresponds to the waste–paper basket that acted as

an ignition source in the Dalmarnock fire test, the heat release rate curve follows an approximately parabolic growth curve, also referred to as the t^2 fire, between 200 seconds and 800 seconds. As the data does not entirely describe a full sofa burn, from ignition to burn–out, it was extrapolated to describe the rest of the burning based on the fact that the total energy stored in the sofa is calculated to have a value of about 379 MJ [4]. As the total heat released by the sofa would have to equal this value, a limited range of possible heat release rate curves for the sofa can be considered. Feeding this information into K–CRISP gave a surprisingly good result as shown in Figure 4.40. The figure shows the compartment average temperatures from the sensor measurements and the K–CRISP outputs. Although the temperature using the low peak heat release rate is under–predicted by about 100°C, all three predictions using preset heat release rate show very similar burn–out time to that of the real fire.

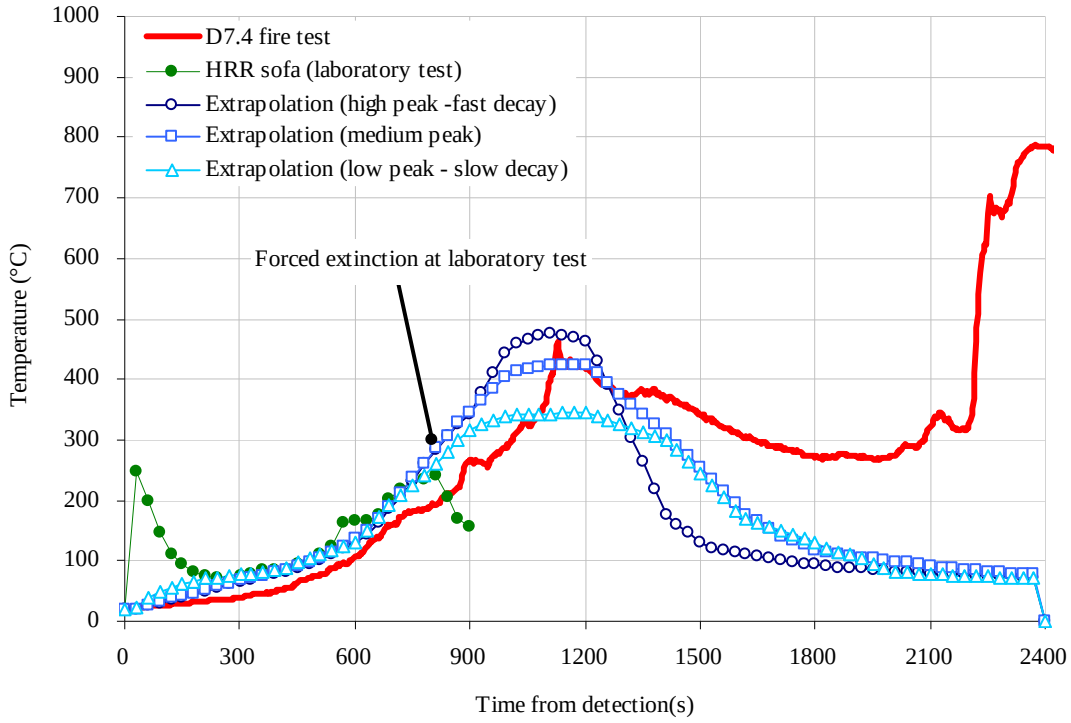


Figure 4.40 – K–CRISP prediction with pre–defined sofa HRR

Prediction only with sensor in fire origin

K–CRISP was then run only using the average compartment temperature of *Room 1*. The change of the PDF for *door status* parameters (see Figure 4.41) shows an interesting result. While the fire origin was easily inferred in this case, the model struggled to infer the status of all doors during the early stage of the steering procedures. As no temperature data of any other rooms, except that of the fire origin, was available, K–CRISP did not have any way to infer the status of the doors in *Room 2*, and *Room 3*, and the exterior door in *Room 3*. The status of the door in *Room 1*, however, could be inferred first even without the temperature data from the *Corridor* because the fire might have died if the door had been closed, while the actual fire grew near to 500°C suggesting that there was enough oxygen in the room. For the same reason, the PDFs for the other doors also gradually converged to "open" as the model observed that the fire kept growing rather than dying due to lack of oxygen.

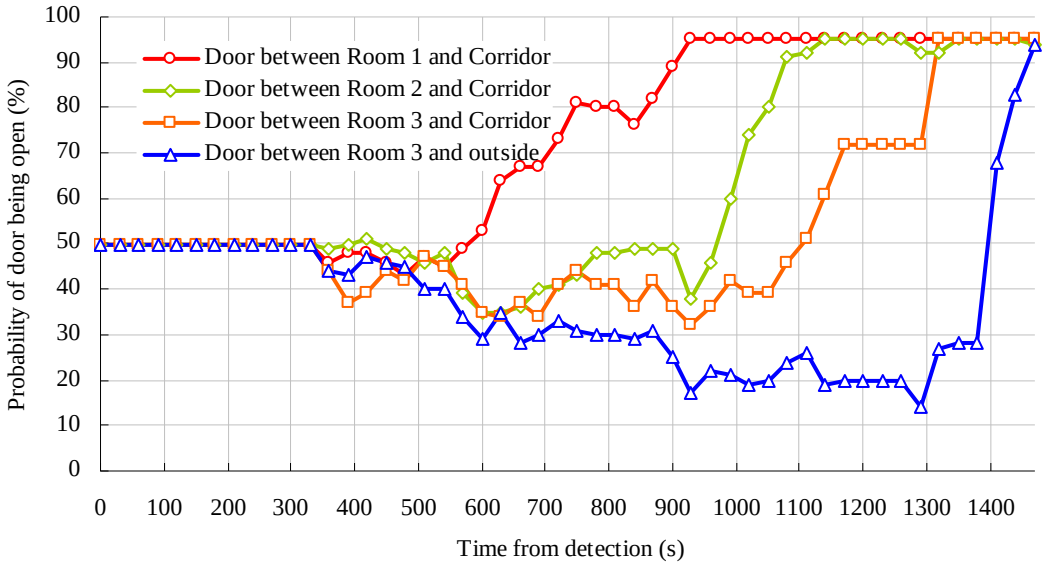


Figure 4.41 – Change of PDF for *door status* when only sensor data from *Room 1* is used

The evolution of the predictions was relatively more accurate, in terms of the fire room temperature, than other previous cases, as the model could focus on a single sensor data in this case, i.e. the average temperature of *Room 1*. Unlike the results in the previous case (see

Figures 4.38 and 4.39), the average temperature of *Room 1* matched very well while the temperatures of all other rooms were slightly under–predicted.

Prediction with a single faulty sensor

Finally, K–CRISP was tested with a single channel of faulty data included in the sensor observations. As mentioned in the FireGrid D7.4 demonstrator results in chapter 2, one of the thermocouple sensors to measure the vent–flow temperature at the doorway between *Room 1* and the *Corridor* (channel 12) was measuring significantly lower temperatures than other neighbouring sensors throughout the fire test, though it was not excluded from the steering procedure during the live test as the temperatures were still within the acceptable range which had been defined in advance. Fortunately most of the vent–flow temperature sensors were given relatively small scaling factors during the live test as they were expected to be relatively less accurate than the room gas temperature considering the conventional limitations of zone models; thus it did not affect the prediction significantly. In the realistic case, however, the same 'failure' could potentially occur in any other sensors and a single faulty set of data has the potential to ruin the overall prediction.

In this case, several tests were carried out with different values of scaling factor given to channel 12. Figure 4.42 shows the result when it is given the same scaling factor as other thermocouple sensors for room temperature. The figure shows that the gas temperature in *Room 1* gradually diverges from the sensor measurement because the model was trying to match the vent–flow temperature of channel 12, which is close to the ambient temperature. The quality of the predictions strongly depended on the scaling factor given to the faulty channel. The model was still able to keep up with the sensor observations even with slight under–prediction during the *growing* phase if a relatively small scaling factor is given. But if channel 12 was too much emphasized, e.g. having the same scaling factor as other channels, the entire prediction was ruined because the nearly ambient temperature of channel 12 forced

the entire prediction to be lowered significantly throughout the fire test. This confirms that a single faulty sensor channel could potentially affect the overall prediction; thus, the capability of filtering out such sensor channels is essential in order to achieve accurate predictions.

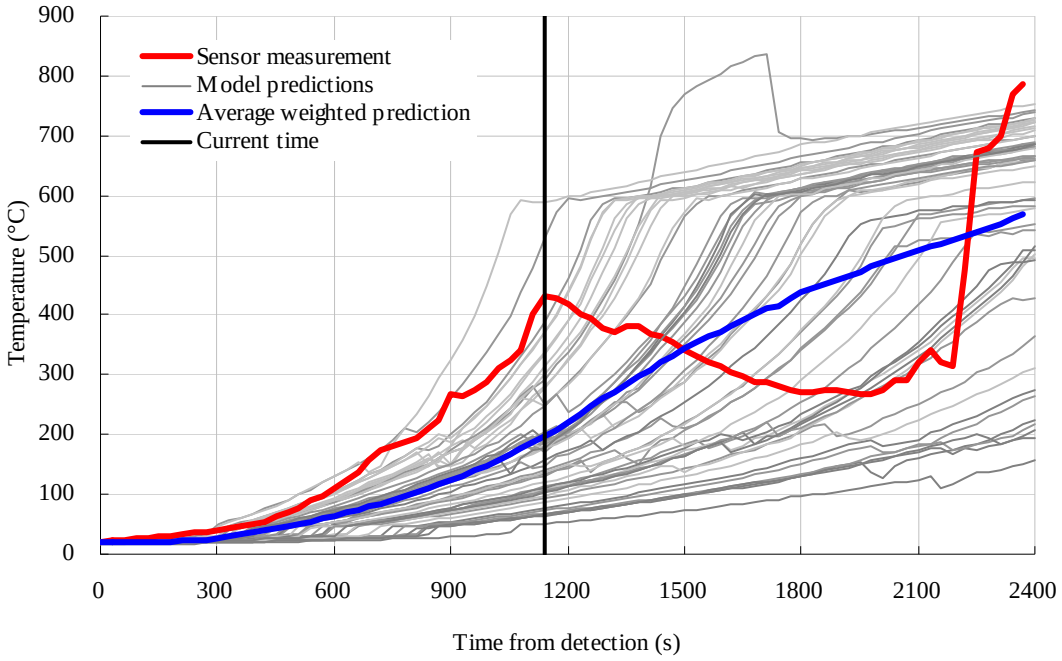


Figure 4.42 – K–CRISP prediction at 1140 s with a single faulty data included

4.5 Application of sensor-linked model to Dalmarnock fire test

As mentioned in the introduction section of this chapter, the main purpose of the FireGrid D7.4 demonstrator was to assess the performance of K–CRISP, thus the model was also developed partially taking that specific test into account, in terms of the scale or the duration of a fire which the model is going to simulate. Although the model was designed and the input parameters were intentionally defined to be randomized as much as they reasonably can be, it might therefore have inadvertently been developed in a way that the model works

better for a specific fire case, e.g. the D7.4 demonstrator. It was therefore determined that the model performance should also be tested with the Dalmarnock fire test data with the same setup which had been used when the model is tested with the D7.4 demonstrator data. The prediction results are shown in this section; however, the initial condition of the model could not be entirely the same as the D7.4 demonstrator case because the two tests were very different in terms of the sensor instrumentation.

All the rooms were equipped with sensors in the FireGrid D7.4 demonstrator, however, only the fire compartment was heavily instrumented during the Dalmarnock fire test. Therefore, a single average temperature of the compartment (see Figure 2.7, page 21) is used as the 'sensor data' in the latter test. The predictive capability of K–CRISP in terms of the fire origin and the door status is not considered here as no sensor data is supplied from any other rooms and a similar case has already been investigated in the previous section (see Figure 4.41).

While the fire behaviour of the D7.4 demonstrator was complex in terms of the unexpected change of the *phase*, the fire of the Dalmarnock test was relatively simple but had a tricky temperature rise in the early stage, not conforming to a general fire growth, i.e. t^2 shape. Moreover, it was already shown in a previous section that K–CRISP was not able to accurately reproduce the fire behaviour of the Dalmarnock test if a single item is used, which indicates the importance of the fire phase definition as it defines the point in time when the second item is ignited. Thus the predictive capability of the model, in this case, will clearly depend on whether the fire *phase* can be defined properly.

Prior to the full simulation of K–CRISP, the fire *phase* was first tested using the compartment temperature in order to be assured that the model can define the *phase* accurately also with the Dalmarnock case. Figure 4.43 shows the fire phase definition based

on three different temperatures, i.e. the maximum, the average, and the minimum temperature of the Dalmarnock test compartment.

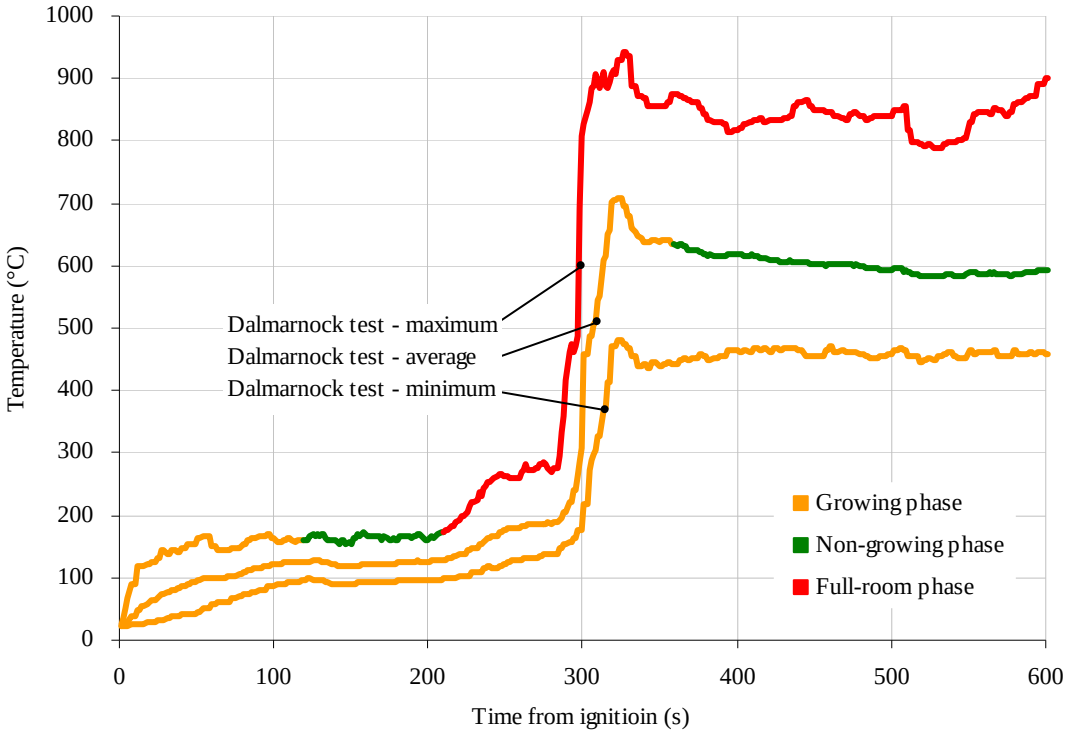


Figure 4.43 – Fire phase definition of the Dalmarnock fire test data

When K–CRISP used the maximum temperature for its fire phase definition, it was possible to accurately define the phase transitions as shown in the figure. It changed the phase from *growing* to *non–growing* at about 120 seconds from detection when a slight decrease in the temperature was detected. Then the rapid temperature rise at about 220 seconds triggered the *full–room* phase and kept it for the rest of the simulation. Unfortunately, the current algorithm of changing the phase from *growing* to *non–growing* did not work well with the average temperature and the minimum temperature. Because the temperature did not show enough decrease during the levelling off period, K–CRISP tended to keep the phase to *growing*. In such case, K–CRISP will not be able to trigger the secondary item; thus it will generate scenarios only by using a single item, thus resulting in a poor match.

Nonetheless, if K–CRISP defines the fire phase properly, e.g. by using the temperature data which shows the fire development clearly enough to distinguish “phases”, it was able to predict the fire development with reasonable accuracy, as it had been using the D7.4 demonstrator data.

Chapter 5

Discussion

5.1 Introduction

As described in chapter 4, successful integration between sensors and a fire model enabled some degree of forecast of fire development. The sensor–linked model, K–CRISP, was found to be able to adjust its input parameters based on the real–time sensor measurements and it was possible to generate predictions which can be “steered” and tend to become gradually closer to the real fire. This chapter provides further analysis of the prediction results and discussion about the limitations of the methodology in terms of choice of fire modelling approach, capabilities of the fire model, i.e. zone model, randomizing input parameters, matching criteria, optimising steering, and predictive capability.

5.2 Choice of fire model

A fundamental decision of strategy in the development of the simulation tool is the basic type of model to be adopted for the fire representation. Various approaches to integrating measurements and predictions of fluid dynamic phenomena are widely used in other disciplines, e.g. data assimilation techniques used with CFD in weather forecasting. The potential advantage is the much greater level of detail available in the predictions, but the drawbacks in terms of computational demands and complexity of implementation severely outweigh these and would render super real–time predictions generally impractical for real fires.

On the other hand, while zone modelling has many advantages in terms of flexibility, there are various inherent limitations. Some of these are in common with those widely recognised in application of zone models, e.g. inability to correctly represent fire phenomena in environments, particularly large spaces, where the two–layer approximation breaks down, and the challenge of correctly representing fire behaviour where local gas flow effects significantly influence the burning behaviour, e.g. in the neighbourhood of ventilation openings. The most obvious consequence of choosing a fire model representation which describes parameters averaged over large control volumes, i.e. the zones, is the practical challenge in integrating these with the detailed measurements which are spatially varying. Thus, while thermocouples are likely to have noticeably different values at every individual measurement point, the zone model can only represent two layer temperatures (i.e. the hot layer and the cold layer) which may not even be a good approximation to the broad distributions of temperature, thus there is an inevitable mismatch between model and test in terms of spatial detail.

While there is no reason why the respective values cannot still be compared, i.e. just selecting hot or cold layer values from the model according to location, thereby simplifying

the handling of sensor channels, this will only work well when the hot and cold layers can be adequately distinguished in the test. But even in such case, there will be a sudden temperature rise in the model outputs when the layer interface crosses over the location of the sensor, as shown in Figure 4.25 (page 88), and this can potentially affect the predictions.

As there was no significant stratification in the FireGrid D7.4 demonstrator, at least beyond the very early stages of the fire, we have chosen a different approach by ignoring layers in the test and simply averaged all the thermocouple data in each room, irrespective of height, and compare it with predicted hot layer temperature. K–CRISP was able to generate accurate prediction even by using a single average temperature value from each room; however, this approach may give a mismatch in other cases when the upper layer is relatively shallow, or when there is a considerable temperature difference between layers, e.g. fire in an atrium. More sophisticated modelling treatments, e.g. multi–zone models or explicit representations of stratification, might provide an advantage in such circumstances. Nonetheless, the current strategy may still be a more practical approach for real applications where it will not be possible to locate thermocouple trees in the middle of each room.

An obvious benefit of selecting a zone modelling basis for the simulation tool is computational flexibility. In accordance with this, it is noted that the computational demands of the live simulations of the demonstration test described here were fairly easily catered for; while the test itself was fairly modest in scope, the HPC resources actually called upon were equally modest, extending to a tiny fraction of the overall capabilities of the relevant hardware, and could have been increased without difficulty. Nevertheless, to guarantee efficient use of the K–CRISP code for more complex scenarios would certainly require further exploration of some of the computational issues—in particular it may be necessary to parallelise the ‘‘scenario selection’’ or the “steering” part of the code to enable it to handle

cases requiring more rapid turnaround of the results assessment with larger numbers of scenarios.

5.3 Limitations of fire model

The fire model of K–CRISP assumes that the fire radius will increase at a uniform rate, which if the pyrolysis rate per unit area remained roughly constant, would tend to give an approximately t^2 fire growth curve. It is important to note that many fires form to t^2 but equally many may not.

As the fire in the FireGrid D7.4 demonstrator showed roughly a t^2 growth (see Figure 2.16, page 31), K–CRISP was able to generate fire scenarios of which the temperature during the *growing* phase matched quite well with the sensor measurements. The fire of the Dalmarnock fire test, on the other hand, did not follow a t^2 growth in early stage, when the fire was started at the waste–basket and rapidly spread to the sofa due to the accelerant poured into the ignition source. In fact, the accelerant caused the fire to grow very fast in the first 100 seconds from ignition, raising the compartment average temperature from 23°C to over 100°C in about 50 seconds (see Figure 2.7, page 21). In spite of the rapid temperature rise, K–CRISP was still able to generate a few scenarios with steep fire growth in the very early stage and it was capable of steering the parametric space of relevant parameter, thus generating more ‘fast growing’ fires in the next iteration. This was possible because the initial parametric space of those input parameters related to the *growing* phase was set wide enough to let the model come up with such extreme cases during the generation of first set of scenarios. Soon after the average compartment temperature reached about 120°C, however, it levelled off for the next 100 seconds until the bookcase was ignited, leading the fire to flashover. As the fire growth up to this point was not t^2 shape at all, K–CRISP struggled to

generate optimum scenarios; thus most of them were given very small weighting factors during this stage. Moreover, as the model still tried to match its t^2 fires to the sensor measurements, all the predicted fires had various temperatures at the "current" point, resulting in the model struggling more to come up with the right point in time when the secondary ignition or the 'flashover' occurs. This indicates that no matter how flexibly the input parameters for the model are set the generated scenarios are constrained by the capability of the fire model being used.

The results of the application of K–CRISP to the Dalmarnock fire test data however showed that the model is still capable of simulating a similar fire growth, even if it is not exactly a t^2 growth, if the fire phase is defined in a certain way. In terms of 'fire phase', the fire growth in the early stage of the Dalmarnock fire test, where the fire grows and levels off, could be divided into two phases. The rapid growth can fall into the *growing* phase as it did in all the tests, and the flat temperature rise could potentially be defined as the *non–growing* phase, as shown in Figure 4.43 (page 108), if the 'maximum' temperature is used. This will let the model generate fires which are gradually decaying during this period, enabling a better match than a single t^2–growth fire. This will then enable the model to 'detect' the secondary ignition time more correctly and ignite the *full–room* item at the right time. Nevertheless, as this simple solution still depends on the capability of the model to define the fire phase correctly, it may just shift the responsibility to the other side of the model. The use of the fire phase and its limits are discussed in a later section.

During the small tests in the FireGrid D7.4 demonstrator, smouldering fire was intentionally designed and the capability of the sensors to detect the fire was assessed. However, the model was not tested with smouldering fire as it cannot truly simulate the smouldering phase at the moment. Although the fire in the final live test of D7.4 demonstrator was not expected to have a smouldering phase, a simple modification was made to the model in order to be

assured that it can handle the fire even when it grows much slower than expected. A parameter, *delay time*, was added in the fire model of K–CRISP, which simply defines the time until the fire becomes the 'flame fire'. Although the model was able to delay the ignition during the smouldering phase while it was tested with reference data, a true simulation of smouldering phase may be necessary if the model is to be used as a tool for smoke movement analysis.

Finally, K–CRISP is not yet capable of simulating multiple fires. It cannot start fire in more than one room and fire cannot spread to any other rooms. However, it is obvious that the real fire can start in multiple places in the same time and it is common that fire spreads to adjacent rooms or even to remote rooms via ducts or any vertical connections over floors. Enabling the model to simulate multiple fires or the fire spread among rooms may not be very difficult as the same methodology used for defining the secondary ignition time can potentially be applied, i.e. igniting the secondary room based on the probability. The difficulty of predicting the "spreading" time, however, may even be more challenging than the secondary ignition of an item.

5.4 Randomizing input parameters

While the fire growth was relatively slow in the FireGrid D7.4 demonstrator, the temperature initially rose very rapidly in the Dalmarnock fire test as shown in the earlier chapter. The actual values for those input parameters related to the *growing* phase in both tests were laid close to the edge of the parametric space which was defined prior to the test, and the probabilities of the model to select those values were in fact very small. Although the model was able to choose close to the real one in both tests shown in the thesis, there is still some

risk of not having any optimal scenarios if the number of scenarios is not sufficiently big, or if the initial PDFs are not properly set due to user's misjudgement.

A simple revision of the initial PDFs based on the sensor measurements may enable the model to easily overcome this problem. After running numbers of simulations of the live fire tests, it was seen that the parameters which govern each fire phase have a certain relation with the temperature rise. If this link could somehow be known in advance, the model may be able to set its parameter with "probable" values. For example, a very rapid temperature rise shown in the Dalmarnock fire test could be used to estimate the value of an estimated value for the *rate of flame spread* parameter and it was approximately 0.005 m/s. Thus, even if the mean value of the initial PDF for this parameter is erroneously set too far from this value, the model may still be able to adjust the value to be a closer to 0.005 but keeping large variance. In case of the *fuel load* parameter, it may improve the efficiency of the model more significantly because the parameter directly affects the point when the fire goes into the *non–growing* phase. A continuous observation of the current temperature and the total duration of burning may enable a rough estimation of how much fuel has been burnt up to that point. These estimations can potentially be extended to other parameters and eventually improve the efficiency of the model by lowering the required number of scenarios.

5.5 Matching criteria

As mentioned in the description of the methodology it is necessary to examine our assumptions regarding the conditions for the χ^2 distribution. The first assumption, in Equation 10 and 11 (page 44), is that $O_i - E_i$ is normally distributed with variance σ_i^2. Unfortunately, the variance does not depend solely on the accuracy of the sensor measurement, but instead is primarily dominated by the fact that the model will never be a

perfect representation of the real fire. For example, in the Dalmarnock fire tests [4], the range of observed temperatures at different thermocouple locations within the hot layer at any point in time was roughly proportional to the "mean" temperature. A "zone" model will however predict that all locations in the hot layer would be at the same temperature. If the variance was assumed to be solely due to measurement accuracy, then the model could never achieve a "good match" to the observation. For thermocouple measurements we assumed that $\sigma_i^2 \sim T_{layer}$. For other sensors we also had to make similar assumptions about "reasonable" values for the variance. On examining the contributions made by each sensor to the overall value of X^2, if some sensors were making consistently high contributions this may be an indication that we have underestimated σ_i^2 for this sensor.

The other assumption that we made was that all sensor measurements would be independent of one another. Since there is no way in principle of ensuring that sensor measurements are not made in nearby locations, or different readings taken at short time intervals, it follows that there will be a degree of correlation between different measurements. This adds a substantial degree of complexity to the simple χ^2 test described in chapter 3. To make matters worse, the degree of correlation between different observations cannot simply be defined in advance, but would need to be determined at each point in time as the fire developed. This remains as an outstanding problem to be solved in a satisfactorily robust manner.

Due to the above difficulties, our value of the quantity X^2 defined in Equation 10 (page 44) may not follow a χ^2 distribution at all, or not with the expected number of degrees of freedom. We may therefore have too many (or more likely, too few) simulations achieving a reasonable goodness–of–fit to the observations. If there are too many, it is not possible to discriminate between different input parameter values (the likelihood function is a uniform

PDF and hence the posterior distribution for any parameter value is the same as its prior distribution). If there are no matching simulations at all, the situation is the same (the likelihood function is uniformly zero). With just a few "good" matches, the posterior distributions may be very "spiky" or converge too rapidly on an incorrect value. This was the reason for use of the constant scaling factor for each of the X^2 quantities, thus the steering procedure succeeds in giving more weight to simulations (i.e. sets of input parameter values) which achieve better matches to the observations, but the procedure is not as efficient as a true Bayesian inference would be.

A careful thought should also be made on which part of the past data is going to be used during the steering procedure. The model can either use only the most recent set of sensor data, i.e. the data of last 30 seconds, or the entire past observations, i.e. all the data from the ignition to the current point, or all the data in the current fire phase. During the *growing* phase the one end of the temperature line is bound to the ambient temperature at the ignition point and the fire growth approximately follows the t^2 curve; therefore, the overall temperature rise still could be matched even by a narrow comparison range (or even a single point). During the *non–growing* phase, however, the 30 seconds window was not enough to distinguish between two temperature lines with different slopes but with very close values within the comparison range. Thus the model could distinguish optimal scenarios easier when the entire sensor data within the phase was used.

In numerical point of view, however, the enlarging the size of the comparison range can affect the weighting factor as the number of data point becomes bigger. As the goodness–of–fit test results are normalized by the degree of freedom (total number of data points being compared) when the chi–squared distribution is approximated to Normal distribution, this effect should be minimized; however, it was seen that the weight of a scenario becomes smaller when more data is used in the matching procedure. However, a small change in the

goodness–of–fit test, e.g. manipulation of the variance, can potentially resolve this problem in fairly easy manner.

5.6 Optimising steering

Notwithstanding the issues mentioned in the previous section, the use of X^2 as a criterion proved to be a reasonable approach, offering a way to assign a probability for goodness–of–fit for a given number of degrees of freedom and also a method to ascribe more weight to specific sensors. For the test when most of the sensors were involved in the steering procedure, we have put the biggest scaling factor on hot layer thermocouples, with correspondingly reduced factor on more derived parameters such as radiometers. In practice, a minority were excluded from the steering procedure in the post–test simulations, specifically a thermocouple which experienced a fault during the test (but which did not go out of range) and also the gas flow velocities as they can mislead the predictions as shown in the results in the previous chapter. Regarding the latter, it is noted that while temperature is generally a reliable and fundamental reflection of the underlying fire behaviour, there is more uncertainty in velocity parameters. Moreover, with no detailed flow–field modelling the zone model is not well placed to produce accurate local predictions of such, so it is appropriate to reduce their influence. As per discussion above, further mechanisms for improving the balance between the different sensor inputs are in principle available via adjustment of the individual variance values during the simulation, i.e. reducing them when parameters become known with more confidence, and vice versa, but this strategy has only been exploited to a limited extent in the thesis and its use still requires further exploration.

Numerical issues may also arise in the steering procedures. The updating procedure compares sensor data and the model outputs, adjusting the PDFs according to the parameter

values which had been used in generation of scenarios which were a closer representation of the real fire. For example, when no relation exists between the parameter values and the goodness–of–fit, the PDFs ought to be left unchanged. However, because there are a finite number of scenarios, which may not always be sufficient to inform this decision, the code might sometimes change the PDFs when it ought not to as shown in Figure. Defining the fire phase and releasing only the PDFs which are related to that phase could assist in mitigating this problem and it will certainly increase the computational efficiency of the model in all these cases. But on the other hand, there are some obvious issues in how accurately the model can define the fire phase with fluctuating or pulsating fires, and how many scenarios are really needed to adequately address these more complex fire behaviours. Nevertheless, as long as the fire phase is defined correctly, it could certainly reduce the number of required scenarios quite significantly, there being no need for the model to generate scenarios to prove that the current fire status has no connection with effectively redundant parameters.

Further computational advantages result from the flexibility of the chosen modelling methodology, which can easily incorporate direct measurements where these are available, thereby decreasing the load on the inference procedure. For example, door positions can be updated if sensors can supply this information, and models of structural response to the fire are naturally assisted if measurement information on component temperatures, or even deflections, is available. In the extreme case, it is also possible to allow partial override of the model predictions by the inputs of a human expert.

Finally, it is worth noting that the model could in principle be used to support decisions on the required sensor infrastructure, including choice of which devices are most useful and their required resolutions, i.e. how many are needed and where they should be located. An encouraging finding of the current study was that the evolution of the predictions can still be fairly accurate even with a single representative hot layer thermocouple value.

5.7 Predictive capabilities

The fire behaviour on the day of the FireGrid D7.4 demonstrator was more complex than expected, giving an initial appearance of heading off towards flashover, followed by a period when it seemed to be burning out and the final sudden transition to flashover. In fact, this rather unexpected behaviour proved an excellent test of the model capabilities and good basis for discussing the potential benefits and limitations of such a system.

First of all, the results shown in K–CRISP predictions (Figures 4.11 to 4.16, page 76) demonstrate that the lead time of reliable predictions varies significantly during the test. Suggestions that the fire may proceed to flashover obtained in the period up to 1200 seconds are based on the assumption that there is enough fuel so that the fire can grow up to ventilation controlled phase. Since the main combustible item, the sofa, was identical with the one used in the Dalmarnock tests and the laboratory tests had shown that it started to burn out around 1200 seconds from ignition [4] as shown in Figure 4.40 (page 103), a similar time to commencement of the decay phase might also have been reasonably assumed for the current test. Passing this information to the model, it showed the possibility to represent a set of predictions consistent with this. However, in real fires, such information is typically unlikely to be available and the worst case is of far more general interest to the end user. Hence, in the current methodology, the parameter for fuel load is simply fixed to a reasonably large value, by default, such that the model's capacity for reaching flashover is never limited by the predicted commencement of the burnout.

The predictions, however, are quickly revised when the fire actually begins to decay, as indicated by the changing shape of the "prediction" curve. At this stage, a finite possibility of flashover occurrence is still retained in the outputs supplied to the end user, reflecting other uncertain phenomena which cannot be modelled with any confidence. In fact, the initial predictions are not necessarily inaccurate, as they do correctly reflect the potential escalation

of the incident, and it is clear that the fire came very close to a flashover transition around this initial peak; indeed, the compartment temperatures were slightly lower when the actual flashover finally began, at around 2200 seconds. However, just before the latter time, the dominant assessment of the model was that the fire would continue to decay, and there is very little advance warning on the transition itself. This deficiency derives from the lack of any explicit representation of the ignition of, or flame spread to, the second major "burning item" in the model (and any additional items beyond that). To include such details would render the modelling procedure vastly more complex, but for little apparent benefit as potential fire development and flame spread is inherently highly unpredictable in environments with a multiplicity of materials and irregular surface geometries (requiring, for example, detailed information on material composition and temperature–dependent thermal properties). Moreover, if modelling accuracy is dependent on the precise location and orientation of burning items, there is an additional burden in measuring and defining these, and inevitable uncertainties concerning whether they are still *in situ* at the time of the incident, etc.

In the sensor–linked model shown in this thesis, therefore, inferences have been made about secondary ignition by continuously observing the sensor data to detect any significant temperature rise, rather than trying to explicitly predict the fire spread using an extension of the fire model. This was shown to work reasonably well and the system was apparently able to respond to the change of the fire phase sufficiently quickly. Nevertheless, the predictive capability of the model strongly depends on the accuracy of the fire phase definition; therefore, the fundamental constraint on predictive capability must be considered and allowed for in the information provided to the end user, i.e. though the model is suggesting that the fire is probably burning out, recognition of a finite potential for a sudden flashover transition must be retained and highlighted, as appropriate.

The model's prediction can also be affected by faulty sensor data as shown in Figure 4.42 (page 106). As such sensor failure can commonly occur in the real fire situation, the capability of identifying and filtering out a specific erroneous sensor channel is required. The methodology used in this thesis, i.e. excluding the sensor values when the measurements lie outside the predefined range, worked reasonably well and was able to exclude some faulty sensor channels in time; however, it failed to exclude the sensor channel 12 which gave a significantly lower temperature than other adjacent sensors but which was high enough to be within the physically bounded range. A more sophisticated filtering algorithm can in principle be developed which not only excludes sensors with obviously erroneous or unphysical values but also compares each data with those from neighbouring sensors and excludes (or puts less scaling factor on) the specific channel if it is not "consistent" with others. However, this would require great care in order to avoid excluding reliable data which varies from expectations due to some unexpected artefact of the fire behaviour or other unforeseen phenomenon.

Notwithstanding these limitations, it is clear that the current steering approach can be very effective overall in directing the evolution of the fire parameters, thereby, in general, driving the fire predictions towards the measurements. This procedure offers a great advantage over blind selection approaches where the parametric space of the fire scenarios remains constant throughout the incident. Comparisons of simulations results with and without adjusting PDF model parameters for steering confirms that the former is greatly more efficient, particularly as the duration of the simulation lengthens, with the subset of scenarios which are a reasonable match to the current measured fire conditions eventually becoming dominant. This contrasts the more primitive selection approach with no PDF refinement, where the proportion of good matches actually diminishes as the fire behaviour becomes more complex. The availability of probabilistic information in conjunction with the predictions additionally facilitates the generation of output which has been interpreted to be more informative to

potential end users, i.e. providing both an indication of potential future hazards and an estimate of the degree of confidence of the predictions. It may also be used to support decisions on direct intervention strategies, e.g. the use of sprinklers, or to direct egress by means of intelligent signage or way–finding systems. As the smoke movement is one of the critical factors in terms of safe egress, the model's capability of inferring fire origin and door status can provide useful information to such systems.

However, these are not trivial issues, and it may not simply be a matter of examining the current model predictions, which may be constrained by various uncertainties, but other factors will often need to be considered. In general, it is best to err on the side of conservatism, i.e. wherever it is assessed that there is a chance of hazardous conditions evolving, which exceeds a certain very small threshold, end users should be informed of this danger. All these aspects require further exploration and discussion in conjunction with fire and rescue service personnel.

Chapter 6

Conclusions and further works

6.1 Conclusions

An example of a sensor–integrated fire model, K–CRISP, has been established. The model played a key role as a simulation component in the FireGrid prototype emergency response system. A zone modelling basis has been adopted which provides a practically useful approach to the problem, and the randomization aspect of the model together with the Monte–Carlo approach has been extended in order to achieve sufficient flexibility of the model. Standard deviation and the goodness–of–fit test based on chi–squared were chosen for matching criteria to distinguish the scenarios which fit well with sensor observations. Bayesian inference was then used to feed this “knowledge” into the model, enabling it to focus on the optimal scenarios in the next iteration. The predictive capability of the sensor–

linked model was tested using data from two full–scale fire experiments, i.e. the Dalmarnock fire tests and the FireGrid D7.4 demonstrator, in pseudo real–time and in pseudo live mode. The system has been implemented on HPC systems and uses other procedures, in particular the concept of fire phases and bypass of the inference procedure for any parameters directly measured, to limit the computational demands of the model, thus permitting real–time forecasting of a simple multi–compartment scenario with very modest computational resource. In addition to the fire behaviour, egress and structural collapse are also included in the predictions.

The application of K–CRISP to a live fire test data using the best–match scenario based on the scenario selecting procedure showed that it was possible to provide super–real–time forecasts of the fire development in a conceptually simple way. The model was capable of selecting a “best” scenario which fits sufficiently well with the sensor observation up to the current point in time; however, this prototype version of the model was not robust enough in terms of efficiency as it kept generating scenarios irrelevant to the real fire due to the absence of any “learning” process. It was also difficult to forecast critical events such as flashover as a single best–match scenario was not capable of representing the overall trend of the fire development in other well–matched (but not the best) scenarios.

Implementation of a steering procedure based on the goodness–of–fit test, on the other hand, offers a significant advantage over simpler scenario selecting approaches based on a best–match scenario. It can significantly increase the efficiency of the model as it generates more scenarios using the optimal parameter set; thus more scenarios lie close to the sensor measurements. For the cases examined it was shown to be possible to adjust the parametric space of the model’s input and to gradually steer the model towards the real fire condition. Summarizing all the scenarios together with their “weight” information enabled some degree

of predictive capability of the fire development and of critical events such as flashover and structural collapse.

The model was also capable of inferring information such as fire origin and the status of the door positions in the compartment. When fairly abundant information was supplied, i.e. average temperatures of all rooms, the model could predict the fire origin and the door status quickly and accurately even in the very early stage of the fire development. The prediction of the average compartment temperatures of all rooms in this case also matched well with the sensor measurements. But even when no sensor measurements from the fire origin room were supplied, the model was still able to infer that information. This is more encouraging in terms of the practical use of the model because a real building may not have sensors in all rooms and the model may have to infer such information just by using sensor observations from adjacent rooms.

However, it was shown that stream of erroneous sensor data could potentially ruin the overall prediction if a sophisticated way of excluding such sensors from the steering procedure is absent. As it is not easy to distinguish between faulty data and reliable data in some cases, developing an algorithm for such a filter would require careful thought. The current definition of the fire phase also showed some limits in defining the phase correctly. This is important because the fire phase is one of the most essential factors for achieving accuracy in the model's predictions.

In summary, the results of this thesis using a prototype sensor–linked model, K–CRISP, and the application to the full–scale compartment fire tests demonstrate that the predictions can evolve in a way that closely follows the conditions of a realistic fire. The overall methodology facilitates the generation of confidence parameters associated with the various future scenarios, which might be important information for potential end users. Although

there are inevitable limitations on the value of the predictions, and how they should be interpreted due to the complexity of the fire phenomena, this approach does have interesting scope for supplementing real–time measurements in fire emergencies so as to provide better information about the possible evolution of the various hazards, which ultimately might be of benefit to emergency responders.

6.2 Further works

Further analysis and refinements are required in order to improve the prototype emergency response system presented in the thesis. The following is a priority list in roughly descending order of further works to be done.

Investigation to identify the distribution type of input parameters

All input parameters were assumed to follow Normal distributions in this thesis. Some parameters, however, are known to follow Log–Normal distributions. Although the type of distribution may not critically affect the steering process, it is still vital to use the correct input distribution as the required values at wings could potentially be eliminated in wrong distributions.

Sensitivity analysis on each input parameter

A numbers of input parameters are used to generate a single scenario and it is obvious that the influence of each parameter may not be the same. For example, a small change of a certain parameter value can significantly change the hot layer temperature while others have negligible influence. Focusing on those “important” parameters may increase the efficiency of the steering procedure and may even allow the model to ignore some parameters if they are not affecting the prediction at all.

Improved filtering algorithms for faulty sensors

One channel conspired to defeat the filtering procedure; thus an improved algorithm is required for filtering or ranking confidence in sensor channels. Comparing each sensor measurement with those of neighbouring channels could be one possible solution. Simpler procedures such as the scoring method used in gymnastics, i.e. dropping the lowest and highest, could also be useful.

A robust way of defining fire "phase"

In this thesis, the fire phase was defined based on simple characteristics of the fire compartment temperature, i.e. average temperature, maximum temperature, temperature rise, etc. The fire phase was defined correctly in most cases; however, some results showed that the model may not be able to define the phase correctly if the temperature rise is not clear enough in terms of the condition of each phase. As the fire phase is one of the most essential factors for accurate and efficient prediction, further study to establish a more sophisticated way of defining the fire phase can be very useful.

Use of the model for smoke movement and egress analysis

As shown in the prediction results, K–CRISP was capable of inferring fire origin room and status of doors (open/close) even by using only temperature data from each room. Although human behaviour aspects are not considered in this thesis, inferences related to them were automatically being derived by the model in the course of the fire simulations. Since smoke movement and egress are more of interest in realistic complex buildings further application of a sensor–linked model for analysing these types of scenarios could be very interesting.

Combining units of K–CRISP for true parallel run

Even though each unit in K–CRISP successfully communicated with other units without any significant delay via "files", a "pure" parallel run could potentially increase the performance

of the model if a sophisticated programming skill using “Message Passing Interface (MPI)” is adopted. The most important aspect would be the model’s capability of dealing with unexpected errors such as an individual calculation becoming stuck.

Development of dedicated fire model for sensor–linked simulation

A zone model, CRISP, was sufficiently flexible and fast to be used as a super–real–time simulation tool. Further improvements, however, could in principle be made to the model enabling it to be even more suitable for this type of use. For example, a multi–zone approach may reduce the inevitable temperature rise when the sensor crosses over two different layers.

References

[1] S. Welch, A. Usmani, R. Upadhyay, D. Berry, S. Potter, J. L. Torero, Introduction to FireGrid, Chapter 1 in The Dalmarnock fire tests: experiments and modelling, G. Rein, Abecassis–Empis, R. Carvel (eds.), 2007, pp.7–29, ISBN: 978–0–9557497–0–4

[2] K. McGrattan, Fire Modelling: Where are we? Where are we going?, Fire Safety Science, volume 8, 2005, pp.53–68, doi:10.3801/IAFSS.FSS.8–53

[3] D. D. Drysdale, An introduction to fire dynamics, 2nd edition, John Wiley and Sons, Chichester, UK, 2002, ISBN: 0–471–97291–6

[4] G. Rein, C. Abecassis–Empis, R. Carvel (eds.), The Dalmarnock fire tests: experiments and modelling, ISBN: 978–0–9557497–0–42007

[5] G. Rein, J. L. Torero, W. Jahn, J. Stern–Gottfried, N. L. Ryder, S. Desanghere, M. Lázaro, F. Mowrer, A. Coles, D. Joyeux, D. Alvear, C. A. Capote, A. Jowsey, C. Abecassis–Empis, P. Reszka, Round–robin study of a priori modelling predictions of the Dalmarnock Fire Test One, Fire Safety Journal, volume 44, 2009, pp.590–602, doi:10.1016/j.firesaf.2008.12.008

[6] W. Jahn, G. Rein, J. L. Torero, A-posteriori modelling of Fire Test One, Chapter 11 in The Dalmarnock fire tests: experiments and modelling, G. Rein, C. Abecassis–Empis, R. Carvel (eds.), 2007, pp.193–210, ISBN: 978–0–9557497–0–4

[7] J. K. W. Wong, H. Li, S. W. Wang, Intelligent building research: a review, Automation in Construction, volume 14, 2005, pp.143–159, doi:10.1016/j.aucon.2004.06.001

[8] A. V. Gheorghe, D. V. Vamanu, Adapting to new challenges: IIDS for emergency preparedness and management, International Journal of Risk Assessment and Management, volume 2, issue 3-4, 2001, pp.211–223, doi:10.1504/IJRAM.2001.001506

[9] N. A. Theophilopoulos, S. G. Efstathiadis, Y. Petropoulos, ENVISYS environmental monitoring warning and emergency management system, Spill Science and Technology Bulletin, volume 3, issues 1-2, 1996, pp.19–24, doi:10.1016/S1353-2561(96)00023-0

[10] S. Tufekci, An integrated emergency management decision support system for hurricane emergencies, Safety Science, volume 20, 1995, pp.39–48, doi:10.1016/0925-7535(94)00065-B

[11] G. Napolitano, L. See, B. Calvo, F. Savi, A. Heppenstall, A conceptual and neural network model for real–time flood forecasting of the Tiber River in Rome, Physics and Chemistry of the Earth, volume 35, 2010, pp.187–194, doi:10.1016/j.pce. 2009.12.004

[12] J. Mu, X. Zhang, Real–time flood forecasting method with 1–D unsteady flow model, Journal of Hydrodynamics, Ser. B, volume 19, 2006, pp.150–154, doi:10.1016/S1001-6058(07)60041-9

[13] P. Jain, M. C. Deo, Real–time wave forecasts off the western Indian coast, Applied Ocean Research, volume 29, 2007, pp.72–79, doi:10.1016/j.apor.2007.05.003

[14] F. Bouttier, P. Courtier, Data assimilation concepts and methods – March 1999, Meteorological Training Course Lecture Series, ECMWF, Reading, 2002

[15] M. Leblanc, A. Trouvé, Inverse modelling of Enclosure fire dynamics. In proceedings of the 6th U.S. national combustion meeting, 2009

[16] A. Cowlard, L. Auersperg, J. B. Richon, G. Rein, S. Welch, A. Usmani, J. L. Torero, A simple methodology for sensor driven prediction of upward flame spread, Turkish Journal of Engineering and Environmental Sciences, volume 31, issue 6, pp.403–413, 2007

[17] A. Cowlard, Sensor and model integration for the rapid prediction of concurrent flow flame spread, PhD thesis, The University of Edinburgh, 2009, http://www.era.lib.ed. ac.uk/ handle/1842/2753

[18] W. D. Davis, T. Cleary, M. Donnelly, S. Hellerman, Using sensor signals to analyze fires, Research and Practice: Bridging the gap, Fire Suppression and Detection Research Application Symposium, Proceedings of Fire Protection Research Foundation, 23–25 January, Tampa, FL, pp.205–224, 2002.

[19] W. D. Davis, Studying the response of building systems to fire using a virtual cybernetic building test–bed, Research and Practice: Bridging the gap, Fire Suppression and Detection Research Application Symposium, Proceedings of Fire Protection Research Foundation, 22–24 January, Orlando, FL, pp.412–421, 2003.

[20] C. Park, P. A. Reneke, M. A. Galler, S. T. Bushby, W. D. Davis, Enhancement of the virtual cybernetic building test–bed to include z zone fire model with HVAC components, NISTIR 7414; p.91, 2007.

[21] W. Jones, R. Peacock, G. Forney, and P. Reneke, CFAST – Consolidated model of fire growth and smoke transport, Version 6, Technical report, 2005

[22] I. Foster, C. Kesselman, The Grid: Blueprint for a new computing infrastructure, Morgan Kaufmann, 1999

[23] D. E. Leidner, G. Pan, S. L. Pan, The role of IT in crisis response: Lessons from the SARS and Asian Tsunami disasters, Journal of Strategic Information Systems, volume 18, 2009, pp.80–99, doi:10.1016/j.jsis.2009.05.001

[24] L. E. Link, The anatomy of a disaster, an overview of Hurricane Katrina and New Orleans, Ocean Engineering, volume 37, 2010, pp.4–12, doi:10.1016/j.oceaneng. 2009.09.002

[25] A. S. Usmani, Y. C. Chung, J. L. Torero, How did the towers collapse – a new theory, Fire Safety Journal, volume 38, pp.501–533, doi:10.1016/S0379–7112(03)00069–9

[26] R. O. Carvel, A. N. Beard, P. W. Jowitt, D. D. Drysdale, Variation of heat release rate with forced longitudinal ventilation for vehicles fires in tunnels, Fire Safety Journal, volume 36, pp.569–596, doi:10.1016/S0379-7112(01)00010–8

[27] L. Han, S. Potter, G. Beckett, G. Pringle, S. Welch, S.–H. Koo, G. Wickler, A.

Usmani, J. L. Torero, A. Tate, FireGrid: An e–infrastructure for next–generation emergency response support, article in press, corrected proof, available online 1 July 2010, Journal of Parallel and Distributed Computing, doi:10.1016/j.jpdc.2010.06.005

[28] P. Reszka, C. Abecassis Empis, H. Biteau, A. Cowlard, I. Fletcher, A. Fuentes, M. Gillie, S. Welch, Experimental layout and description of the building, Chapter 2 in The Dalmarnock fire tests: experiments and modelling, Rein, Abecassis–Empis & Carvel (eds.), 2007, pp. 31–62, ISBN: 978–0–9557497–0–4

[29] C. Abecassis Empis, P. Reszka, T. Steinhaus, A. Cowlard, H. Biteau, S. Welch, G. Rein, J. L. Torero, Characterisation of Dalmarnock Fire Test One, Experimental Thermal and Fluid Science volume 32, 2007, pp.1334–1343, doi:10.1016/j. expthermflusci.2007.11.006

[30] K. McGrattan, Fire Dynamics Simulator (Version 4) – User's Manual, NISTIR 6784, 2003

[31] S. Kumar, G. Beckett, J. Holden, G. Pringle, S. Potter, S. Welch, D7.4: Final Demonstrator Report, FireGrid technical report FG08/004/00, DTI Project Number: TP/3/DSM/6/I/16191, rev2 issued 13, August 2009, http://www.see.ed.ac.uk/_firegrid/ FireGrid/Project_papers/WP7/D7.4/FG08_004_00_final_demonstrator.pdf)

[32] W. Jahn, G. Rein, J. L. Torero, The effect of model parameters on the simulation of fire dynamics, in Proceedings of the 9th International Symposium, International Association for Fire Safety Science, Fire Safety Science, volume 9, 2008, pp.1341–1352, doi:10.3801/ IAFSS.FSS.9–1341

[33] J. N. Fraser–Mitchell, An object–oriented simulation (CRISP 2) for fire risk assessment, in Proceedings of the 4th International Symposium, International Association for Fire Safety Science, 1994, pp.793–804, doi:10.3801/IAFSS.FSS.4–793

[34] J. N. Fraser–Mitchell, Modelling human behaviour within the fire risk assessment tool CRISP, Fire & Materials, volume 23, 1999, pp.349–355, doi:10.1002/(SICI)1099–

1018(199911/12)23:6<349::AID–FAM710>3.0.CO;2–3

[35] J. N. Fraser–Mitchell, CRISP (Computation of Risk Indices by Simulation Procedures) user manual

[36] G. Heskestad, Fire plumes, flame height and air entrainment, The SFPE Handbook of Fire Protection Engineering (3rd edition), DiNenno P.J. (ed.), National Fire Protection Association, Quincy, MA 02269, 2002, chapter 2–1

[37] C. L. Tien, K. Y. Lee, A. J. Stretton, Radiation heat transfer, The SFPE Handbook of Fire Protection Engineering (3rd edition), DiNenno P.J. (ed.), National Fire Protection Association, Quincy, MA 02269, 2002, chapter 1–4

[38] S. J. Melinek, The distribution of fire load, Fire Safety Journal, volume 20, pp.83–88, doi:10.1016/0379–7112(93)90013–G

[39] C. Chatfield, Statistics for technology: a course in applied statistics (3rd edition), Chapman and Hall/CRC, Wiley, Boca Raton Fl, 1983, ISBN–0–412–25340–2

[40] U. Cortes, M. Sanchez–Marre, L. Ceccaroni, Artificial intelligence and environmental decision support systems, Applied Intelligence, volume 13, issue 1, 2000, pp.77–91, doi:10.1023/A:1008331413864

[41] G. Wickler, G. Beckett, L. Han, S.–H. Koo, S. Potter, G. Pringle, A. Tate, Using simulation for decision support: lessons learned from FireGrid, in Proceedings of the 6th International Information Systems for Crisis Response and Management (ISCRAM) conference, J. Landgren, S. Jul (eds.), Gothenburg, Sweden, May 10–13, 2009, http://www.iscram.org/ISCRAM2009/papers

[42] R. Upadhyay, G. Pringle, G. Beckett, S. Potter, L. Han, S. Welch, A. Usmani, J. L. Torero, An architecture for an integrated fire emergency response system for the built environment, in Proceedings of the 9th International Symposium, International Association for Fire Safety Science, Fire Safety Science, volume 9, 2008, pp.427–438, doi:10.3801/IAFSS.FSS. 9–427

[43] Eurocode 3: Design of steel structures, Part 1.2: General rules, Structural fire design,

EN 1993–1–2: 1992, CEN/BSI

[44] J. N. Fraser–Mitchell, S.–H. Koo, S. Welch, Sensor–linked simulation for emergency response, Proceedings of the 3rd International Conference on Human Behaviour in Fire, Cambridge, 13–15 July 2009, pp.561–566