

Sensor-steered fire simulation

SUNG-HAN KOO ^a, JEREMY FRASER-MITCHELL ^b and STEPHEN WELCH ^a

^a*BRE Centre for Fire Safety Engineering, The University of Edinburgh, Edinburgh EH9 3JL, UK*

^b*BRE Global, Bucknalls Lane, Garston, Watford, Hertfordshire WD25 9XX, UK*

ABSTRACT

A sensor-linked modelling tool for live prediction of uncontrolled compartment fires, K-CRISP, has been developed in order to facilitate emergency response via novel systems such as FireGrid. 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. 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.

KEYWORDS

modelling, compartment fire, sensor, steering, Bayesian inference, emergency response

1. Introduction

In fire safety engineering, computer simulation has been widely used for various purposes such as risk assessment, smoke movement analysis and inferring the cause of the fire after the incident. Due to the complexity of fire, however, 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 can lead to the fire developing in a totally

different way. Related work on application of fire modelling tools to full-scale tests involving real fuels provides a useful context [1]. In these studies, detailed simulations were performed both *a-priori* and *a-posteriori*, predominantly using Computational Fluid Dynamics (CFD) models. The results of the blind modelling demonstrate the enormous challenge of predicting the evolution of conditions in complex realistic scenarios [2]. But the results of *a-posteriori* CFD modelling further reinforce the difficulty of obtaining a good match to the conditions of the test, even when the burning behaviours of the individual items involved in the fire had been reasonably well characterised [3,4]. This confirms that without some sort of real-time steering of fire models, achieving a useful forecast capability will remain an insurmountable challenge in many practical circumstances [5].

Modern buildings are rich in sensors which continuously collect data and monitor the building for various purposes such as fire detection, indoor comfort and security. In particular, data from temperature sensors, smoke detectors and movement sensors can provide essential information which might be used to estimate the size and location of the fire or other information about the occupants. Moreover, this information could potentially supply some of the “unknowns” in the fire model and assist in maximizing the accuracy of the resulting prediction. The vision of the FireGrid project [5] is to integrate, or assimilate, real-time sensor data with the output of fire simulation tools in order to provide an estimate of the possible future development of the fire, together with advance warning of extreme events such as flashover and structural failure [6]. Formal procedures for reconciling predictions and measurements, each with a degree of uncertainty, are available via “data assimilation” techniques, but 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.

In this paper, we present a methodology which could underpin a prototype emergency response system exploiting these real-time integrations. The novel aspect of the 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. A live fire demonstration that integrated all the components of the FireGrid project took place at BRE Garston (UK) in October 2008 and is adopted here as case study application.

2. Input parameters

Generally, when running blind fire simulations, most of the unknown input parameters are predefined based on any available information about the scenario, supplemented where necessary by the user’s experience and knowledge of what might be reasonable approximations or assumptions. If a variety of cases are to be considered, e.g. for Monte-Carlo type models for risk assessment, these parameters may be generalised to some extent, but again only within the limits of the user’s expectations or sensible

guesswork. In this study, however, an alternative strategy is adopted where many of the unknowns are progressively inferred based on an assessment of how well the evolution of the fire is matching the live measurements.

The CRISP code [7] 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 proposed 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 [7,8]. 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 this work, several parameters were added to the randomisation in order to achieve a fuller flexibility for the case of fire development in real buildings, which has many unknowns. The additional parameters were mainly those associated with the “fire model” of CRISP, 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.

3. Scenario evaluation

Assuming that the CRISP model itself has sufficient flexibility, and the input parameters for the fire model allow a sufficiently wide range of fire scenarios to be simulated, then Monte-Carlo sampling will eventually uncover the combinations of model parameters which lead to a good match between the calculated scenario consequences and the sensor observations. A basic premise of the modelling strategy is the expectation that 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 a thermocouple? One way is to calculate a “fractional

match” based on the goodness-of-fit test for each set of predicted sensor readings compared to the actual observations.

If we compute X^2 as in equation (1) below,

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

where O_i are the sensor observations for sensor $i = 1, 2, \dots, n$, and E_i are the corresponding CRISP predictions, then by definition [9] X^2 will follow a χ^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. We compute the goodness-of-fit over a time period (30s) 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, we can approximate the χ^2 distribution with a Normal distribution of mean ν , variance 2ν . This makes it very easy to compute the probability that the value of the χ^2 distribution will be larger than X^2 :

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

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

A simple example may explain this better. Assume that we have 50 channels of sensors. CRISP has generated two different scenarios and the X^2 value of each scenario is 34 and 63 respectively. As mentioned above, we can approximate the χ^2 distribution with a Normal distribution of mean 50 and the variance 100, as there are 50 degrees of freedom in this case. The probabilities of the χ^2 value being bigger than 34 and 63 ($1 - \Phi(-1.6)$ and $1 - \Phi(1.3)$) 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 $X^2 = 63$.

4. Steering procedure

A simple scenario selection procedure was adopted in earlier work using (pseudo) sensor measurements obtained in the full-scale fire test undertaken in a furnished apartment at Dalmarnock [10]. With this method a continuous selection from amongst a multiplicity of scenarios generated in Monte-Carlo fashion was achieved using a standard deviation parameter to characterise the difference between the predictions and the measurements. A more robust and generalised approach, however, was required since the generation of new fire parameters was very inefficient in this initial model, being completely independent of the ongoing selection process. Therefore a procedure for “learning” from the previous model performance is required.

One approach to enable the model to “learn”, i.e. to systematically reduce the parametric space from which new scenarios are sampled, reducing the variance of the ensemble, would be to adopt a Bayesian inference procedure to modify the prior estimates for the input Probability Density Functions (PDFs) in the light of the most recent sensor data, as described below.

Bayes’ Theorem is most often expressed in the following form:

$$P(A | B) = \frac{P(B | A) \cdot P(A)}{P(B)} \quad (3)$$

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 a range of different outcomes, then following the observation of a particular outcome B, we can write Bayes’ Theorem as follows:

$$P(A_j | B_i) = \frac{P(B_i | A_j) \cdot P(A_j)}{\sum_k P(B_i | A_k) \cdot P(A_k)} \quad (4)$$

where the denominator is an alternative way of expressing $P(B)$. If we define the “likelihood function” $L(B/A)$ as:

$$L(B_i | A_j) = \frac{P(B_i | A_j)}{\sum_k P(B_i | A_k) \cdot P(A_k)} \quad (5)$$

then Eq. (5) can be simplified to:

$$P(A_j | B_i) = L(B_i | A_j) \cdot P(A_j) \quad (6)$$

i.e. the posterior probability = the prior probability \times 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 we have a fire in a room, adjacent to a corridor that contains a smoke alarm. Let us further assume 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). We will use the notation A_1 = door open, A_2 = door closed, B_1 = detection, B_2 = no detection. If we have no prior information regarding the state of the door, we assign $P(A_1) = P(A_2) = 0.50$. Substituting into Eq. (5), the likelihood $L(B_1/A_1)$ of detecting the fire if the door is open is $0.50 / (0.50 \times 0.50 + 0.01 \times 0.50)$, which equals 1.96. Hence, from Eq. (6), the inferred probability that the door is open, given that the detector operated, is $1.96 \times 0.50 = 0.98$. 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 $0.50 / (0.50 \times 0.50 + 0.99 \times 0.50)$, which equals 0.67, and hence the inferred posterior probability that the door is open is only 0.33.

In a more realistic and complex building, however, the link between cause and effect will be less straightforward. We cannot simply state, as above, what the likelihood function will be, but instead 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 Eq. (5) can be calculated as

$$L(B_i | A_j) = \frac{N(B_i | A_j) / N(A_j)}{N(B_i) / N_{tot}} \quad (7)$$

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:

$$L(B | A_j) = \frac{\sum_{k=1}^{N(A_j)} \alpha_k(B | A_j) / N(A_j)}{\sum_{n=1}^{N_{tot}} \alpha_n(B) / N_{tot}} \quad (8)$$

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

The strategy chosen for model steering in FireGrid 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 Probability Density Functions (PDFs) to be used for various parameters, and how these PDFs are going to be steered. The CRISP code can handle various types of PDF, such as uniform, integer uniform, binomial, Gaussian, log-normal, arbitrary PDF or CDF, etc. For this implementation in FireGrid, we have used 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 we must ensure 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. 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 we simulate, we 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 we insist that the posterior probability distribution is also a Gaussian, a simple solution presents itself.

Using Eq.(6) and Eq.(8), 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:

$$\mu(x | B) = \frac{\sum_{k=1}^{N_{tot}} \alpha_k \cdot P(x = x_k) \cdot x_k}{\sum_{n=1}^{N_{tot}} \alpha_n / N_{tot}} \quad (9)$$

and the variance is:

$$\sigma^2(x | B) = \frac{\sum_{k=1}^{N_{tot}} \alpha_k \cdot P(x = x_k) \cdot x_k^2}{\sum_{n=1}^{N_{tot}} \alpha_n / N_{tot}} - \mu^2(x | B) \quad (10)$$

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 Eq.(1) would follow a χ^2 distribution with an appropriate number of degrees of freedom. The properties of the χ^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*. The underlying assumptions are examined further in the Discussion (section 9).

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 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 example application in this paper, 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, K-CRISP, 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.

5. Sensor data

All measurement data are passed to the fire model via a database, as per the architecture developed for the FireGrid project [11]. Formatting conventions are adopted for each sensor type, enabling rapid set-up. The

detailed test specifications and measurement data are archived under FireGrid [12]. 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 used in the current study.

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 FireGrid 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.

6. Hazard prediction

If the 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 this work, CRISP 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 (CO and CO₂) 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 distance 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, in collaboration with input from fire-fighters [13,14]. The user interface aspects are beyond the scope of this paper 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 a 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 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.

7. Full-scale fire tests

As a part of the FireGrid project, 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.

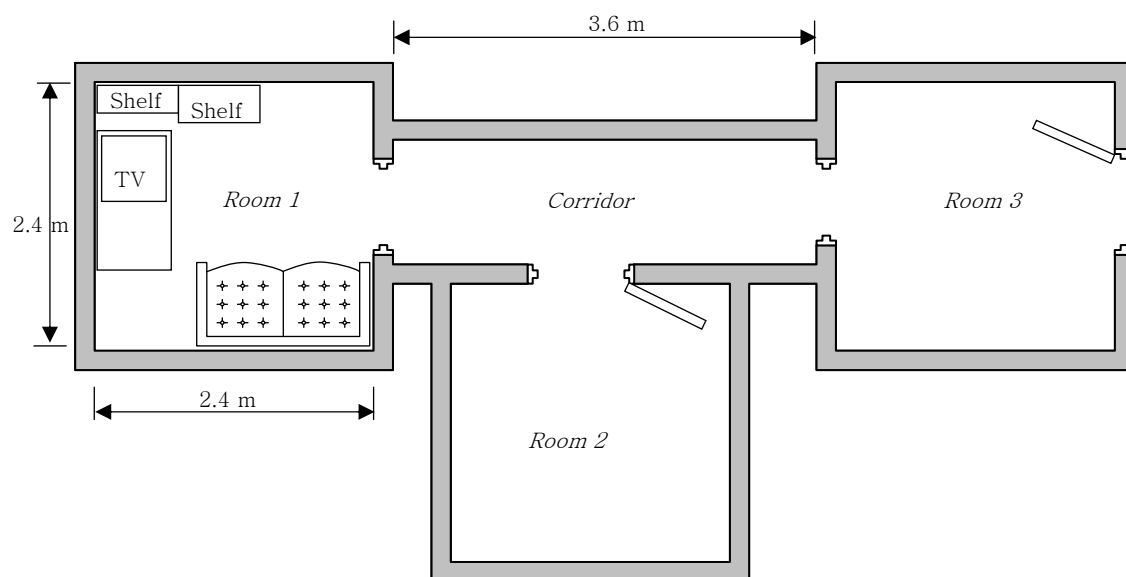


Fig. 1. Plan view of the fire rig and furniture layout for *Room 1* in full-scale demonstration test

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 Fig. 1. 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 m × 1.2 m in

plan and 2.4 m in height. All doors are of size 0.8 m × 2.0 m.

The furniture layout in *Room 1*, the fire room, is also shown in the figure. The furnishings consisted of a two-seater sofa, 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 [1]). 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.

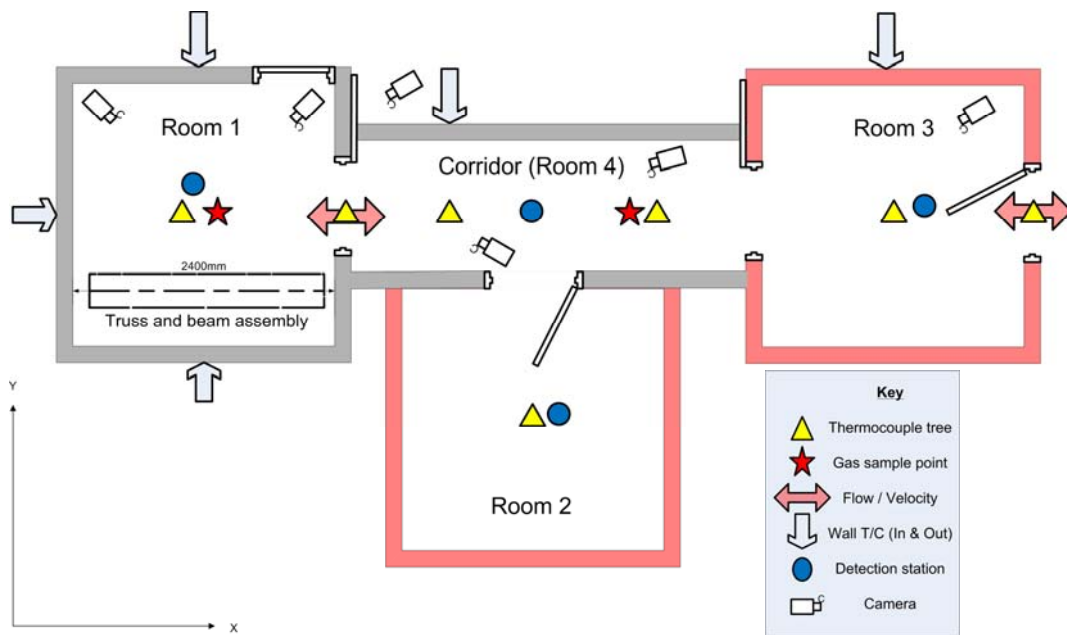


Fig. 2. Rig layout showing locations of instrumentation within the rig for full-scale demonstration test

Fig. 2 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 O₂, CO₂ 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.

The simulations of the fire 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. The K-CRISP code was therefore ported from a Windows PC to HPC resources, specifically the ECDF machine of Edinburgh Parallel Computing Centre (EPCC), known as Eddie, and the national HPCx machine [12-14], 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 required to ensure that a fire model 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, named 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.

8. 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 tests [1], 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 $\sim 10\text{kWm}^{-2}$, eventually leading to flashover in *Room 1*.

While visibility remained acceptable even 4 minutes after the first heat detector was triggered, the fire was of a size that would be challenging to extinguish with a single portable extinguisher (see Fig. 3). Ten minutes after the heat detector was triggered the smoke layer had descended to about 1.2m and remained around this level for a further 10 minutes while the sofa was consumed. 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. 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).



Fig. 3. Four camera view of full-scale demonstration test – 4 minutes after heat detector alarm

K-CRISP performed simulations of this 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 the update, and to come up with new PDFs every 30 seconds which was the sensor data update frequency. Thus 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. Nevertheless, for comparative purposes, all results presented here relate to approximately the same number of scenarios per update as were considered in the live test.

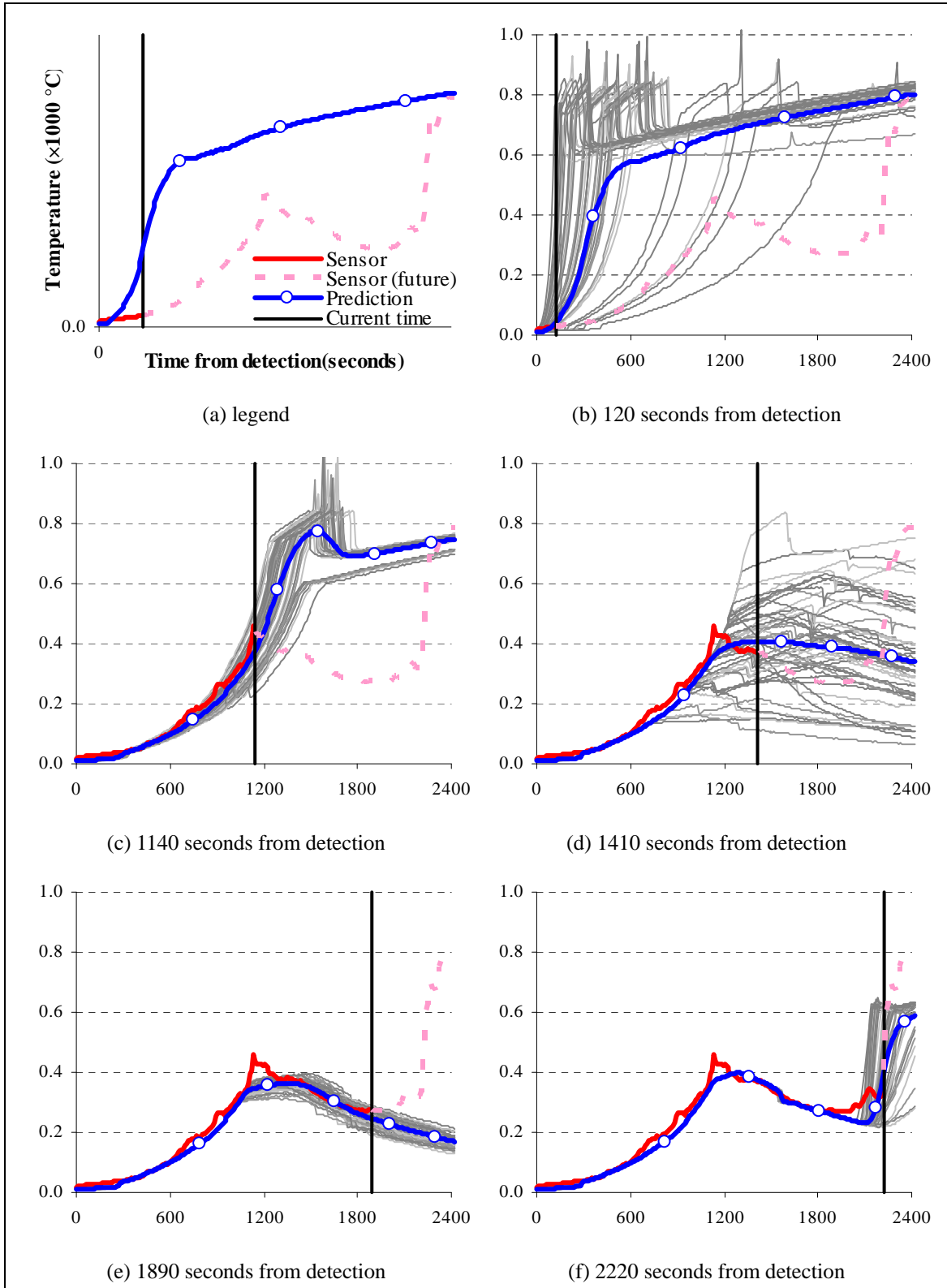


Fig. 4. Comparison between sensor data and K-CRISP predictions for full-scale demonstration test

Fig. 4 compares the hot layer temperature of the sensor measurements and those of the K-CRISP prediction at five different timestamps, with legend indicated in graph (a). The sensor measurements are only “known” up to the current instant in time, as indicated by the thick 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 second period. Finally, the evolution of the “averaged prediction” is depicted by the thick line with symbols.

After several applications of the updating 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. The first series of predictions show that there is a “flashover” (when the temperature exceeds 500°C) around 1300 seconds from detection; 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 decay phase. In the experiment, the temperature continued to decrease until 2100 seconds from detection, as flashover began to develop. 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.

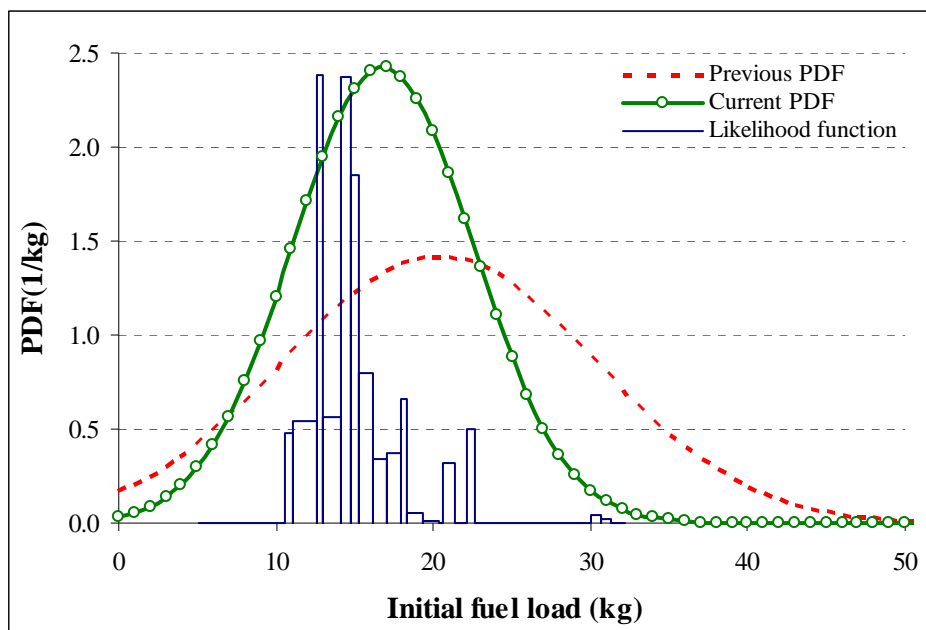


Fig. 5. Example of PDF changes after updating for initial fuel load in the full-scale demonstration test

Fig. 5 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 information is supplied, 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 measurements.

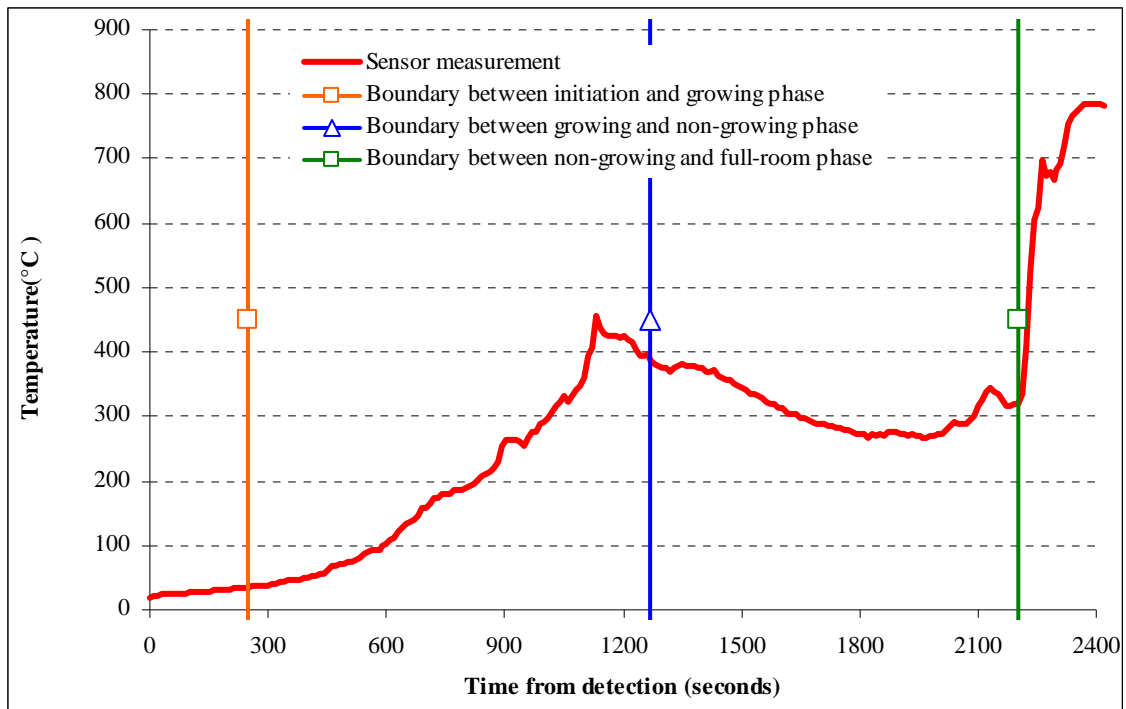


Fig. 6. Hot layer temperature of full-scale demonstration test and fire phase definitions of CRISP

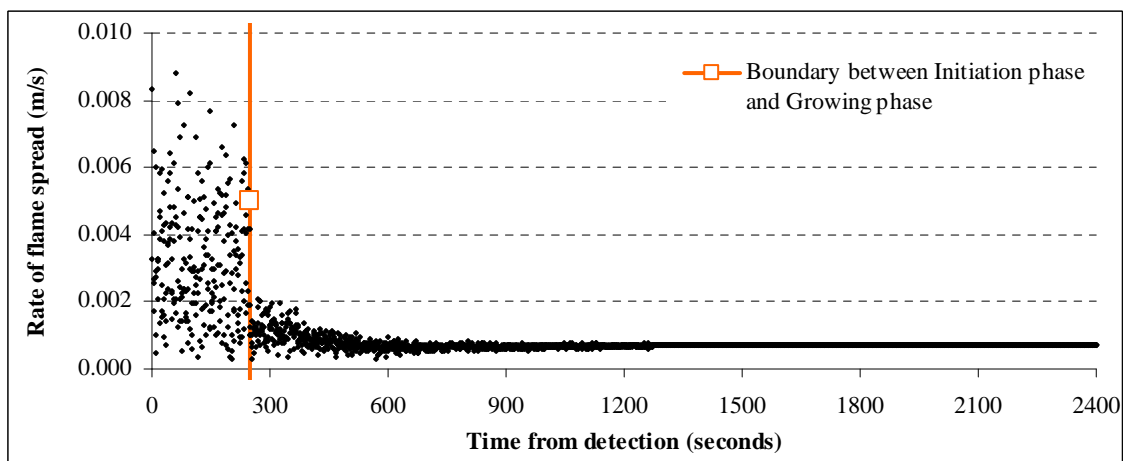


Fig. 7. PDF convergence of rate of flame spread in K-CRISP simulation of full-scale demonstration test

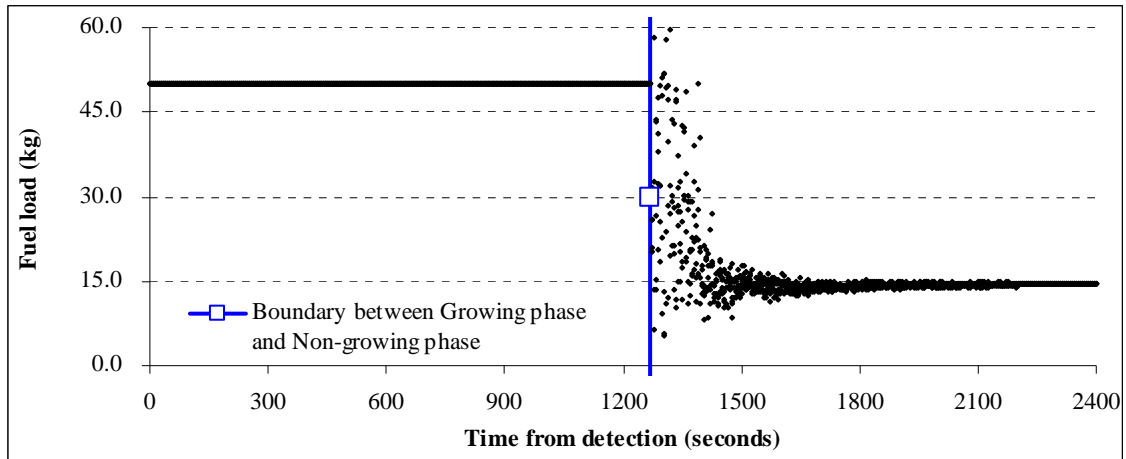


Fig. 8. PDF convergence of fuel load in K-CRISP simulation of full-scale demonstration test

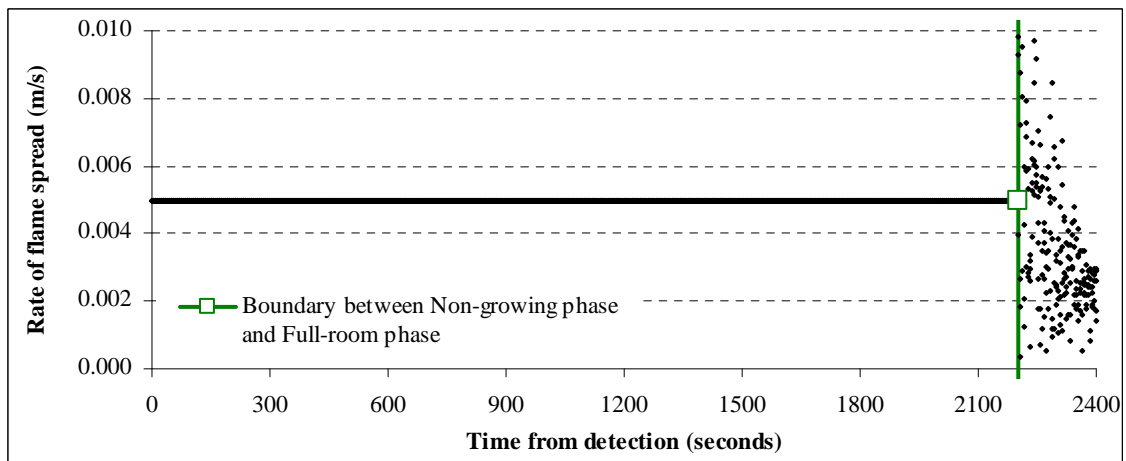


Fig. 9. PDF convergence of rate of flame spread for full room in K-CRISP simulation of full-scale demonstration test

Figs. 6 – 9 present the convergence of a parameter value in another manner, related to the change of the fire phase defined by the model. In Fig. 6, the hot layer temperature from 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 obtained directly from sensor measurements are passed to the model. 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 Figs. 7 and 8). 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, 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 *rate of flame spread* is fixed to its mean value and the *fuel load* is released during this phase (see Fig. 8). 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 didn't have an opportunity to infer the value of the *rate of flame spread*. However, Fig. 9 shows that the model did try to converge the parameter 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 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.

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.* [15].

Finally, the model successfully reproduces the structural collapse, even when the lookup for the member temperature is derived from the thermal model in CRISP rather than from direct measurements. Of course, the case examined is highly idealised, and more realistic scenarios would often involve fire protection materials, but even in those instances it ought to be fairly straightforward to reference information on possible mechanical failure of individual components via this type of procedure. Because the finite element analyses are undertaken in advance, i.e. decoupled from the real-time modelling, there are no significant computational constraints and the value of the output is limited mainly by the level of detail and effort which can be afforded in generating appropriate pre-computed cases.

9. Discussion

9.1 Modelling strategy

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. Another possibility is to ignore layers in the test and simply average all the thermocouple data in each room, irrespective of height, and compare it with predicted hot layer temperature. Whilst this will give a mismatch for shallow upper layers, it may also be a more practical approach for real applications where it will not be possible to locate thermocouple trees in the middle of each room.

Fortunately, there was no significant vertical stratification in the temperature in *Room 1* in the current study, at least beyond the very early stages of the fire. However, this might be an issue in other cases and a different approach might still be required if there are considerable differences in the temperatures, e.g. with a fire in an atrium. More sophisticated modelling treatments, e.g. multizone models or explicit representations of stratification, might provide an advantage in some circumstances, but considering their lack of generality and additional complexity it is likely to be a case of diminishing returns.

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” part of the code to enable it to handle cases requiring more rapid turn-around of the results assessment with larger numbers of scenarios.

9.2 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 Eqs. (1-2), 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 [1], 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 $\sigma_i^2 \sim T_{\text{layer}}$. 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 the previous section. 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 Eq.(1) 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.

9.3 Optimising steering

Notwithstanding the above issues, 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. In this study, 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. 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 flowfield 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 current work 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. 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 was very similar between using a single representative hot layer thermocouple value and an overall average of all the measured values.

9.4 Predictive capabilities

The fire behaviour on the day of the fire test 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 Fig. 4 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 1200s 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 [1], 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 would also have been possible 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 (see below). 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 2200s. 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 this study, 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 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, a recognition of a finite potential for a sudden flashover transition must be retained and highlighted, as appropriate.

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 simulation 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. 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.

10. Conclusions

An example of a sensor-integrated fire model, K-CRISP, has been established to demonstrate a predictive capability for uncontrolled compartment fires which can be “steered” by real-time measurements. A zone modelling basis has been adopted which provides a practically useful approach to the problem, not reliant on a detailed knowledge of combustible items, though there are some consequent constraints in predictive capabilities, in addition to those normally associated with zone models. Implementation of a steering procedure to direct the evolution of the parametric space containing the scenario definitions, amongst an ensemble of models, offers a significant advantage over simpler approaches exploiting “blind” prediction, i.e. those which do not involve any real-time “learning” about the fire conditions. 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.

The results from the full-scale compartment fire test demonstrate that the predictions can be controlled to evolve in a way that closely follows the conditions obtained in a realistic fire. In addition to the fire behaviour, egress and structural collapse are also included in the predictions. The overall methodology facilitates the generation of confidence parameters associated with the various future scenarios, which might be important information for potential end users. In general, near-term predictions can be made with greater confidence than longer term estimates. However, due to the complexity of the phenomena, and the many uncertainties involved, there are inevitable limitations on the value of the predictions, and how they should be interpreted, to a certain extent reflecting genuine uncertainties in the actual fire

development. Nevertheless, 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.

11. Acknowledgements

The work reported in this paper has formed part of FireGrid, www.firegrid.org. This research has been funded by the Technology Strategy Board along with contributions from the other partners in FireGrid. This work has made use of the resources provided by the Edinburgh Compute and Data Facility (ECDF). (<http://www.ecdf.ed.ac.uk/>). The first author gratefully acknowledges the financial support of the BRE Trust, and the additional advice and guidance offered by Prof. José L. Torero at the University of Edinburgh and Prof. Suresh Kumar formerly at BRE Global. The work benefited from discussion with Rochan Upadhyay and many other members of the FireGrid team at The University of Edinburgh, the Edinburgh Parallel Computing Centre (EPCC), and the National e-Science Centre (NeSC).

12. References

- [1] G. Rein, C. Abecassis-Empis, R. Carvel. (eds.) The Dalmarnock fire tests: experiments and modelling, ISBN: 978-0-9557497-0-4, 2007.
- [2] 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* 44 (2009) 590-602. doi:10.1016/j.firesaf.2008.12.008
- [3] W. Jahn, G. Rein, J.L. Torero. A posteriori modelling of Fire Test One, Chapter 11 in *The Dalmarnock fire tests: experiments and modelling*, Rein, Abecassis-Empis & Carvel (eds.), ISBN: 978-0-9557497-0-4, 2007, pp. 193-210.
- [4] W. Jahn, G. Rein, J.L. Torero. The effect of model parameters on the simulation of fire dynamics, *Fire Safety Science* 9: 1341-1352. doi:10.3801/IAFSS.FSS.9-1341, doi:10.3801/IAFSS.FSS.9-1341.
- [5] 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*, Rein, Abecassis-Empis & Carvel (eds.), ISBN: 978-0-9557497-0-4, 2007, pp. 7-29.
- [6] A. Cowlard, W. Jahn, C. Abecassis-Empis, G. Rein, J.L. Torero. Sensor assisted fire fighting, *Fire Technology*, *in press*, available online December 2008.
- [7] J.N. Fraser-Mitchell. An object-oriented simulation (CRISP 2) for fire risk assessment, *Proceedings of the 4th International Symposium*, International Association for Fire Safety Science, 1994, pp. 793-804. doi:10.3801/IAFSS.FSS.4-793
- [8] J.N. Fraser-Mitchell. Modelling human behaviour within the fire risk assessment tool CRISP, *Fire & Materials* 23 (1999) 349-355.

doi:10.1002/(SICI)1099-1018(199911/12)23:6<349::AID-FAM710>3.0.CO;2-3

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

[10] S.-H. Koo, J.N. Fraser-Mitchell, R. Upadhyay, S. Welch. Sensor-linked fire simulation using a Monte-Carlo approach, Proceedings of the 9th International Symposium, International Association for Fire Safety Science, 2009, pp. 1389-1400. doi:10.3801/IAFSS.FSS.9-1389

[11] 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, Proceedings of the 9th International Symposium, International Association for Fire Safety Science, 2009, pp. 427-438. doi:10.3801/IAFSS.FSS.9-427

[12] 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, rev 2 issued 13 August 2009.

<http://www.see.ed.ac.uk/~firegrid/FireGrid/Project_papers/WP7/D7.4/FG08_004_00_final_demonstrator.pdf>

[13] 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 Conference on Information Systems for Crisis Response and Management ISCRAM'09, eds. Landgren J, Jul S. 2009, Gothenburg, Sweden, May 10-13, 2009. <http://www.iscram.org/ISCRAM2009/papers/>

[14] 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.

[15] 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.