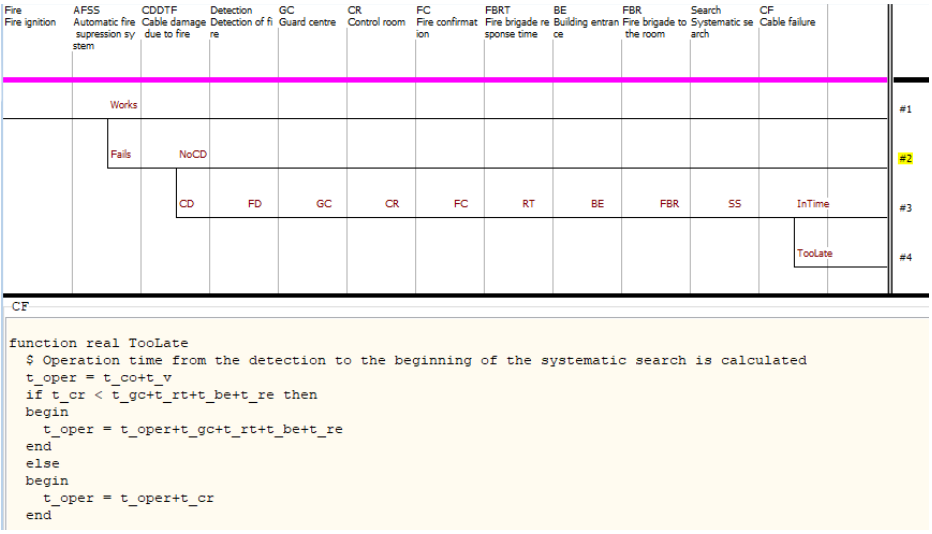
RESEARCH REPORT

VTT-R-01302-19



Analysis of cable fire scenario with simulation-based event tree

Authors: Tero Tyrväinen, Terhi Kling, Timo Korhonen

Confidentiality: Public

Report's title	
Analysis of cable fire scenario with simulation-based event tree	
Customer, contact person, address	**Order reference**
VYR	SAFIR 3/2019
Project name	**Project number/Short name**
New developments and applications of PRA	122529/NAPRA
Author(s)	**Pages**
Tero Tyrväinen, Terhi Kling, Timo Korhonen	22/
Keywords	**Report identification code**
Cable fire, probabilistic risk analysis, event tree, simulation, operation time, firefighting	VTT-R-01302-19

Summary

This report studies probabilistic methods for the analysis of fire risks. In this study, an old nuclear power plant cable room fire scenario has been successfully implemented in a new tool, simulation-based event tree of FinPSA. The main components of the model are Monte Carlo fire simulations and a stochastic operation time model for firefighting. The fire simulations were performed separately using deterministic Fire Dynamics Simulator (FDS), and the results of the simulations were imported to FinPSA. The stochastic operation time model is implemented in FinPSA scripts in eight parts corresponding to different operational phases, including fire detection, guard centre actions, control room actions and fire brigade actions.

The results calculated using FinPSA are approximately same as the results calculated by the old tool, Excel-based Probabilistic Fire Simulator (PFS), but FinPSA offers better model structure, better readability and better maintainability than PFS. Developing complex computation rules is much easier in the script files of FinPSA than it is in the Excel format. FinPSA is therefore considered a useful and practical tool for probabilistic fire risk modelling.

The study may be continued e.g. by more realistic modelling of firefighting, updating the case study for the present needs of PRA-modelling and the latest changes in the control room guidance, analysing failures of multiple cables, modelling fire spreading and analysing uncertainties. The long-term goal is to develop a general-purpose approach for probabilistic risk analysis of fires. It however requires analysis of a wider range of fire scenarios, because other cases may introduce challenges not present in this particular case study.

Confidentiality	Public

Espoo 20.2.2020

Written by	**Reviewed by**	**Accepted by**
Tero Tyrväinen, Research scientist	Ilkka Karanta, Senior scientist	Nadezhda Gotcheva, Research team leader

VTT's contact address

VTT Technical Research Centre of Finland Ltd, P.O. Box 1000, FI-02044 VTT, FINLAND

Distribution (customer and VTT)

SAFIR2022 RG2 members, VTT archive

Contents

1. Introduction

Deterministic fire hazard analyses, investigations of the operating experience at nuclear installations and fire probabilistic risk analysis (PRA) (OECD 2015) have demonstrated that knowledge of the frequency of the occurrence of fires is an important contributor to nuclear power plant (NPP) fire risk assessment. However, fires and associated plant responses are complex phenomena, and therefore the estimates of fire risk are subject to considerable uncertainty. Fire PRA has often been characterized as being less mature and less realistic than other internal events PRA. Perceptions of immaturity can affect stakeholders' use of fire PRA information. Unrealistic fire PRA results could affect fire-safety related decisions and improperly skew comparisons of risk contributions from different hazards (Siu & Sancaktar 2015). Modelling the firefighting defence-in-depth including human actions can be used to reduce conservatism by taking into account the possibility of preventing the damages of critical components and failures of safety functions in a fire situation. Attempts of this kind of approach have earlier been made by e.g. (Hostikka et al. 2012) and (Kloos et al. 2014), but practicality of the method still needs further development.

One of the main limitations of the current fire PRA methodology is that it is not capable of adequately accounting for the dynamic behaviour and effects of fire due to its reliance on the classical PRA methodology i.e. event trees and fault trees (Sakurahara et al. 2014). In this study, simulation-based event trees of FinPSA software (Tyrväinen et al. 2016; VTT, 2019) are used to model a cable room fire scenario. The aim is to develop an approach for more realistic fire PRA. Simulation-based event trees are particularly useful to model impacts of time delays and dynamic dependencies, such as how the time available for fire brigade to arrive depends on the progression of the fire itself, which contains significant uncertainties. The analysis is based on deterministic fire simulations performed using Fire Dynamics Simulator (FDS) (McGrattan et al. 2013). One of the goals of the study is to find the best way to utilise the results of deterministic analyses in the event tree analysis. Combination of deterministic and probabilistic analyses is a general problem common to several areas of nuclear safety, such as the analysis of severe accidents, which has been the main application area of the simulation-based event trees of FinPSA software (Tyrväinen & Karanta 2019) and its predecessor SPSA (Silvonen 2013). Further objectives are also to study the applicability of the FinPSA tool to fire modelling, and identify software development needs.

2. Cable fire case study

This report continues the case study of a NPP cable room fire, which was originally presented in (Hostikka et al. 2012). The analysed cable room contains both power and I&C cables of two redundant subsystems. The cables of the subsystems are physically separated in a multi-level metallic cable tray system. In the places where the cables of different subsystems are close to each other, mechanical shield plates have been installed between the cable trays. The cables are the primary fire load in the room, and the power cables are the most probable source of ignition. In this study, the ignition was assumed to occur in the power cables of subsystem B, and the analysis aimed to estimate the probability of cable failure in subsystem D.

The analysis was divided into two cases: the sprinkler system in operation and the sprinkler system failed. For each of the cases 100 fire simulations were stochastically generated and performed using PFS, i.e. Probabilistic Fire Simulator (Hostikka & Keski-Rahkonen 2003; Hostikka 2008), and FDS, i.e. Fire Dynamics Simulator (McGrattan et al. 2013). For each individual fire simulation, 1000 realizations of stochastic operation time simulations of firefighting operations were created. For each simulation point, it was checked whether the fire brigade was able to suppress the fire before the cable damage. In addition, it was checked whether the firefighting conditions were tolerable when the fire brigade arrived at the room. The model was implemented in PFS, which is an Excel tool, and the stochastic operation time

simulations were performed using random number sampling functions of Excel. The model can be interpreted as an event tree as presented in Figure 1.

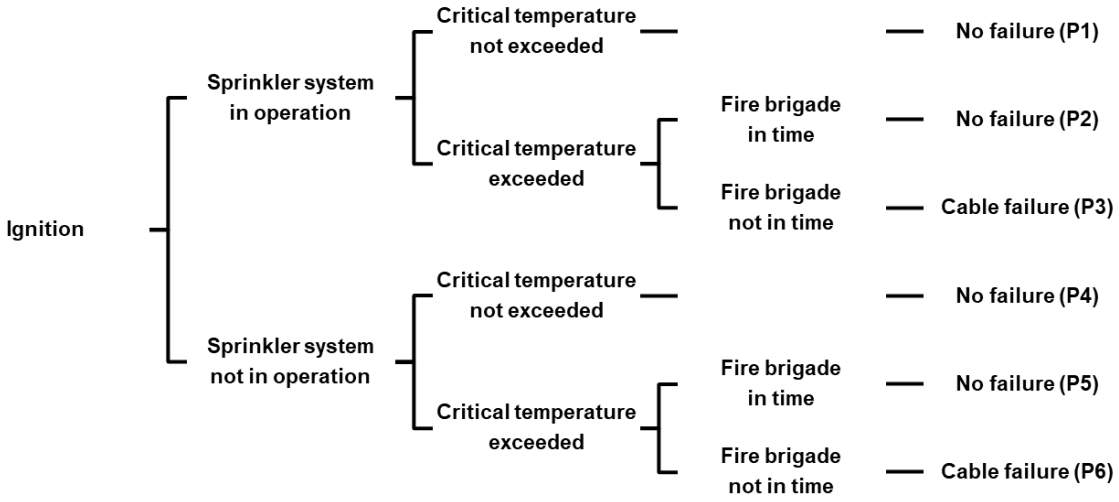


Figure 1: An event tree model for a cable fire (Hostikka et al. 2012).

2.1 Operation time model

The time delays of the operation time model were related to fire detection, control room operations and fire brigade operations. The simulation model includes eight phases:

1. Detection

   There is a delay between the ignition and the detection. Detection can take place through smoke detectors and an automatic alarm system, sprinklers, or through the senses.

2. Alarm

   The information about the detection is transmitted to the security centre and the control room of the NPP as well as to the Emergency Response Centre. The fire brigade receives the information via an alarm system and by phone.

3. Fire brigade response

   After the alarm the fire brigade leaves the fire station and moves to the destination.

4. Fire brigade clearance to the building entrance

   The first assessment of the situation, the unit manager's instructions, and the transition to the front door.

5. Fire brigade arrival at the room of origin

   Finding the destination, moving there and pressurizing the hoses.

6. Co-operation with the plant personnel

   Collaboration is needed between the fire department and the control room. Fire brigade will check the situation and possibly ask for voltage cut-off in the room of origin.

7. (Possibly) Voltage cut-off for the safety of the fire brigade using water-based suppression

   The operator performs the necessary actions from the control room.

8. Systematic search

   Systematic search with thermal imaging. Extinguish the fire when it is found.

Most of the phases consist of multiple actions, and the model includes possible additional time delays caused by human errors and equipment failures. The operational actions are illustrated in Figure 2. After the fire is detected, the control room calls the guard centre, which alarms the fire brigade. There are parallel actions related to control room personnel and arrival of the fire brigade. The control room sends a person to confirm that there is a fire. When the fire brigade arrives, the control room and the fire brigade need to co-operate, and the control room personnel may need to cut-off the voltage before the fire brigade can enter the room and suppress the fire.

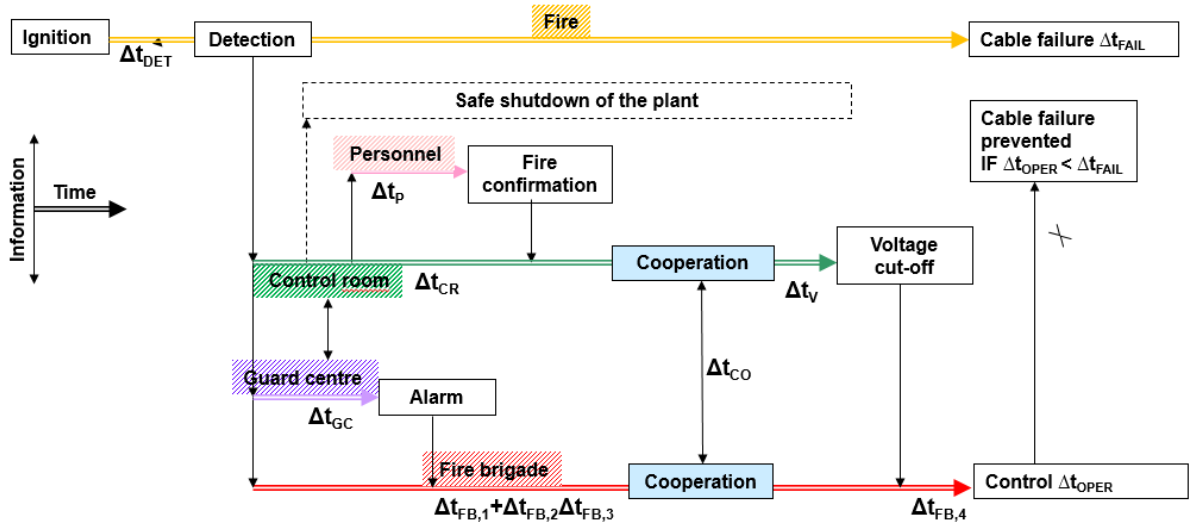


Figure 2: The operational actions related to firefighting (Hostikka et al. 2012).

The total operation time delay from the ignition to the suppression is calculated as

$$\Delta t_{OPER} = \Delta t_{DET} + \max\left[\Delta t_{CR}, \left(\Delta t_{GC} + \Delta t_{FB,1} + \Delta t_{FB,2} + \Delta t_{FB,3}\right)\right] + \Delta t_{CO} + \Delta t_{V} + \Delta t_{FB,4},$$

where Δt_{DET} is the delay from the ignition to the detection, Δt_{CR} is the time it takes for the control room to confirm the fire and perform other preparations before co-operating with the fire brigade, Δt_{GC} is the time it takes for the guard centre to alarm the fire brigade, $\Delta t_{FB,1}$ is the fire brigade response time, $\Delta t_{FB,2}$ is the fire brigade clearing time to the building entrance, $\Delta t_{FB,3}$ is the travel time from the building entrance to the room where the fire is located, Δt_{CO}

is the delay related to the co-operation between the fire brigade and the control room, Δt_V is the time it takes to cut-off the voltage, and $\Delta t_{FB,4}$ is the time it takes to find and extinguish the fire.

The details of the stochastic operation time model are presented in (Hostikka et al. 2012), (Kling 2010) and in Section 4, where the implementation of the model in FinPSA is presented.

2.2 Fire simulations

The Monte Carlo fire simulations were managed by Probabilistic Fire Simulator (PFS) (Hostikka & Keski-Rahkonen 2003; Hostikka 2008). PFS generated the simulation cases using Latin hypercube sampling based on given random variables, created the input files for FDS, managed the simulation runs and performed the post-processing of simulation results automatically. The random variables included the location of the initial fire, the size of the initial fire, properties of power cables and concrete, and the response of the sprinkler system (if working).

For each simulation, it was studied when the temperature of the insulating material around the metal wires reaches the critical temperature. Since there is some uncertainty on what is the critical temperature for a cable to fail, two alternative critical temperatures, 180 °C and 215 °C, were used in the analysis. However, with the sprinkler system working, no cable failures occurred in the simulations. Without the sprinkler system, temperature 180 °C was exceeded in 66 simulations, and temperature 215 °C was exceeded in 64 simulations. The results included also smoke detector alarm time for each simulation. The resulting cable failure time distributions are presented in Figure 3 for the case without the sprinkler system.

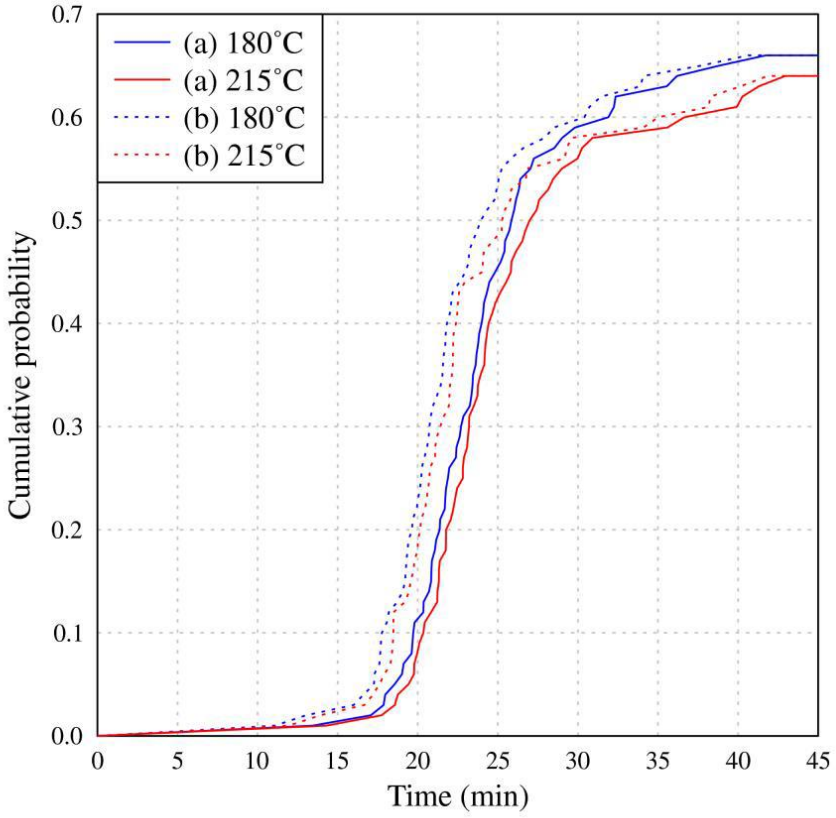


Figure 3: Cumulative probability distribution of cable failure time. The solid lines (a) represent time from the ignition, and the dash lines (b) represent time from the detection (Hostikka et al. 2012).

Firefighting conditions were also evaluated in the simulations. The variables of interest were visibility, temperature and radiation at the doors of the room. For each variable and door, it was determined when the conditions became intolerable for firefighters. Smoke is a barrier to vision, and firefighters are also exposed to heat, which can prevent them from entering the room or at least limit the amount of time firefighters can stay in the room. However, only the information about visibility at door 1 at the altitude of 1.5 meters was utilised in the probabilistic analysis, because it was assumed that the fire extinguishing would primarily be done using this door and the visibility was the most critical variable. Figure 4 shows how the time of intolerable firefighting conditions (visibility at door 1) correlates with the cable failure time (based on the temperature limit 180 °C). In most cases, the visibility was lost around the same time with the cable failure, but there were also cases where the visibility was lost much earlier.

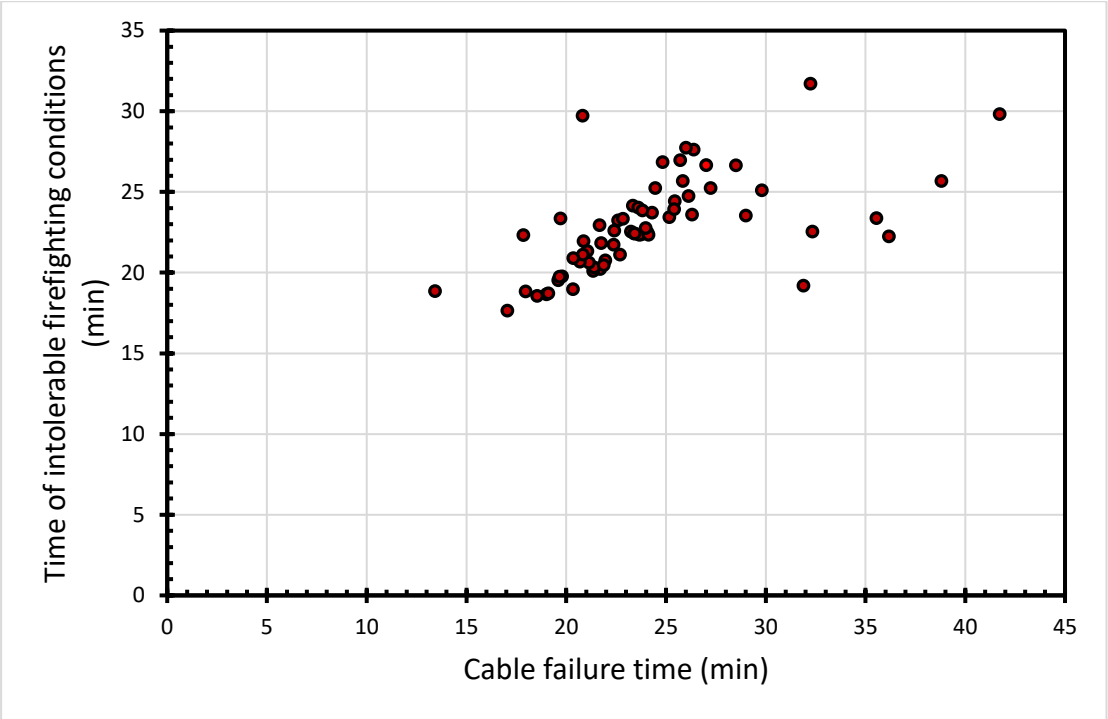


Figure 4: Scatter plot between cable failure time and the time of intolerable firefighting conditions.

Some new analyses were performed for the simulated 100 fires in this study. Figure 5 shows the energy that is released during the fires. The figure shows that the fires can be classified to small or large fires. About 40 % of the fires release less than about 1 GJ of energy and a little more than 50 % release more than 10 GJ. These fires represent different types of fires. Small ones are local fires that do not spread, whereas the large ones represent cable fires that spread along the cable trays (horizontally and/or vertically). Thus, classifying the fires according to the total released energy (GJ) is an option to simplify the event tree model. The small fires did not produce cable damage in the simulations.

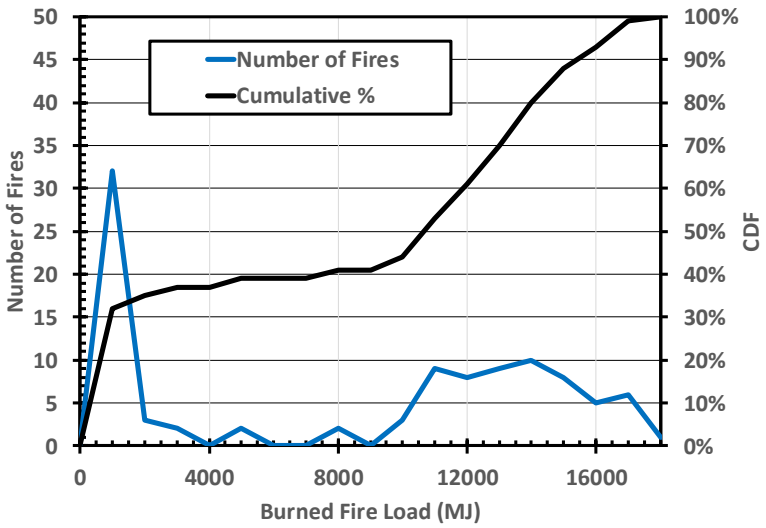


Figure 5. Distribution of the energy released in the cable room fire simulations.

Figure 6 shows the cumulative distribution of detection times for all simulated fires, for small fires (< 1 GJ) and for large fires (> 1 GJ). Lognormal fits are also shown. It can be seen that the total released energy in the fire seems to correlate with the detection time. The fires are (automatically) detected at the early stages and at this point, it does not matter if the fire is or is not spreading along the trays at the later stages of the fire. Thus, just the total heat released during the fire cannot be the reason for the different detection times. Figure 7 shows the detection time compared to the z coordinate of the initial fire. It can be seen that almost all small fires (< 1 GJ) are initiated at the cable trays that are close to the ceiling, while larger fires (> 1 GJ) are initiated at lower cable trays. The detection time of the small fires does not vary very much. The fires are initiated close to the ceiling and, thus, the fire gases and smoke are readily transported to the ceiling mounted detectors. The detection time spread is much larger for the large fires and there is not so noticeable trend with respect to the z coordinate of the initial fire.

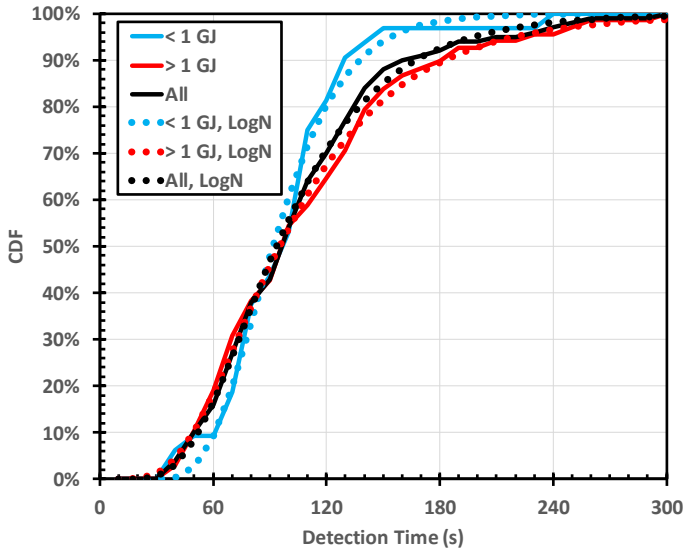


Figure 6. Distribution of detection times. Data is shown for all 100 fires, for small (< 1 GJ) and large (> 1 GJ) fires together with lognormal fits to the data.

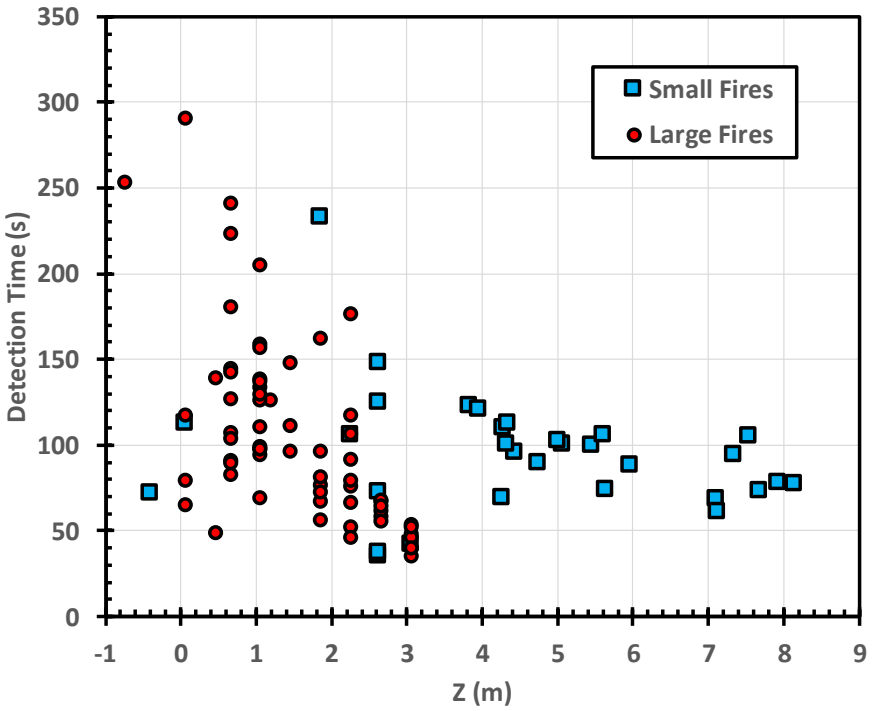


Figure 7. Detection time of the fires compared to the z coordinate of the fire origin.

The small vs. large fire classification of the fires can be further examined. Figure 8 shows the distributions of the peak times of the fires. The peak time is the time when the heat release rate of the fire is at its maximum. It is seen that the peak time of the large fires does not differ much from the peak times of small fires. So, there is not a strong correlation between the peak time and the ability of the fire to spread along the trays. The peak time is a variable that describes how fast the initial fire (user input) grows in the fire simulations.

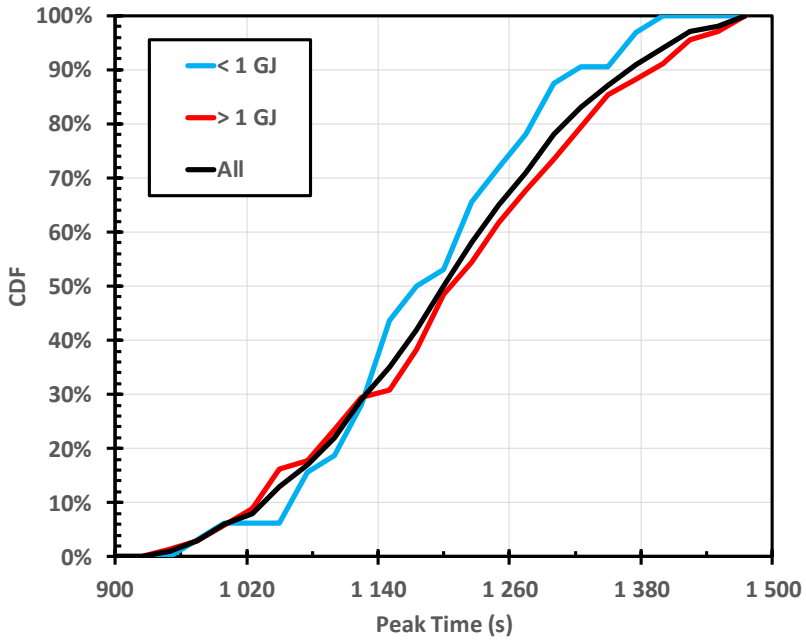


Figure 8. Distribution of the peak time. Data is shown for all 100 fires, for small (< 1 GJ) and large (> 1 GJ) fires.

The above analysis can be used to simplify the event tree based modelling of the cable room fire scenario. The small fires can be treated with a separate, very simple, event tree branch, because small fires do not produce damage to cables on the other redundancy group. The larger fires (> 1 GJ) are treated with a more complicated event tree model. Figure 9 shows that there is not a strong correlation between the peak time and detection time. This implies that the event tree model of the large fires can be simplified by separating the detection time from the simulated fires. The detection time can be implemented in the model as an independent distribution and not a part of the results of a specific fire simulation.

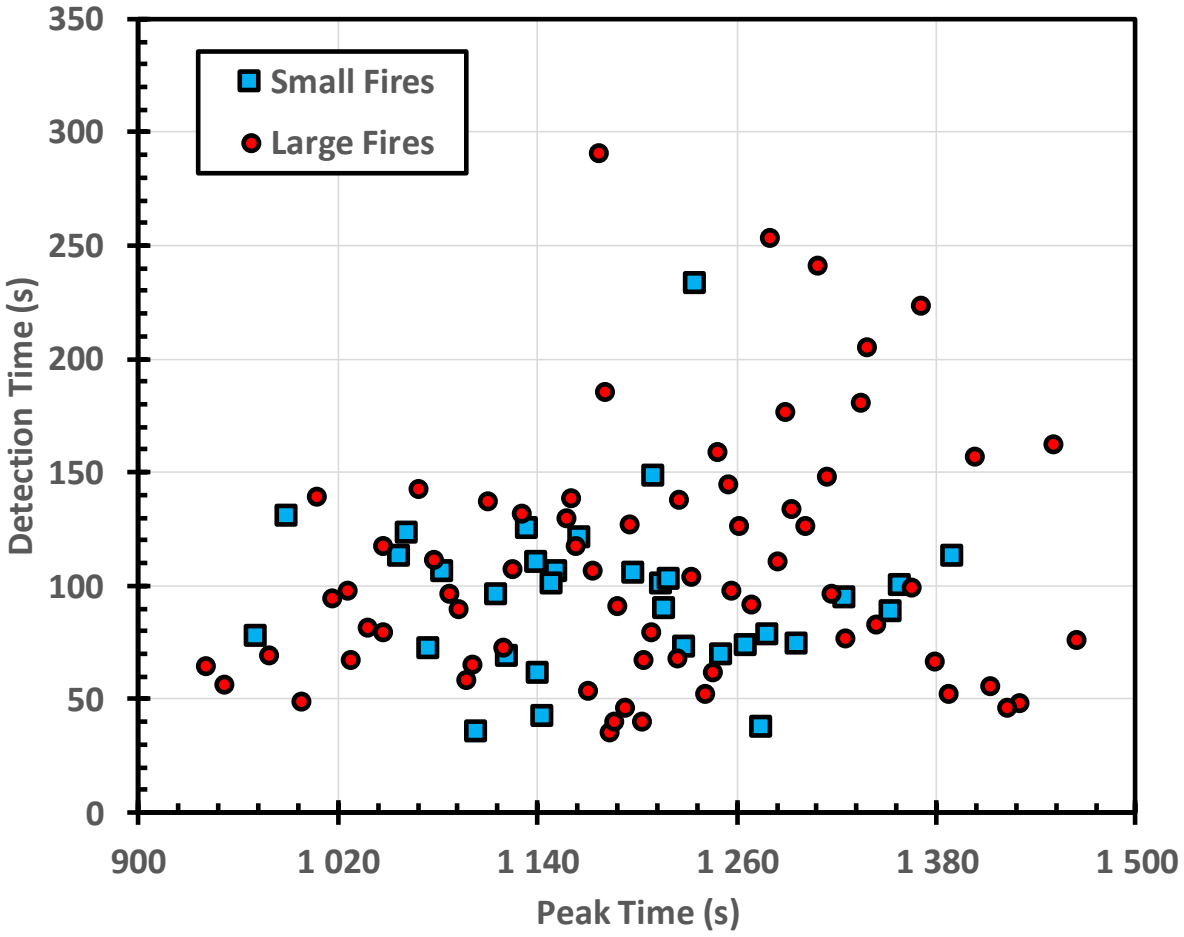


Figure 9. Peak time vs. detection time.

Figure 10 shows the correlation between the damage time and initial fire peak time. It is seen that on the average the damage occurs later for fires with late peak time. Cable damage happens a little bit later than the time when the initial fire has reached its peak value, on the average. This is a quite natural correlation. Fires that grow fast can form damage faster than slowly growing fires.

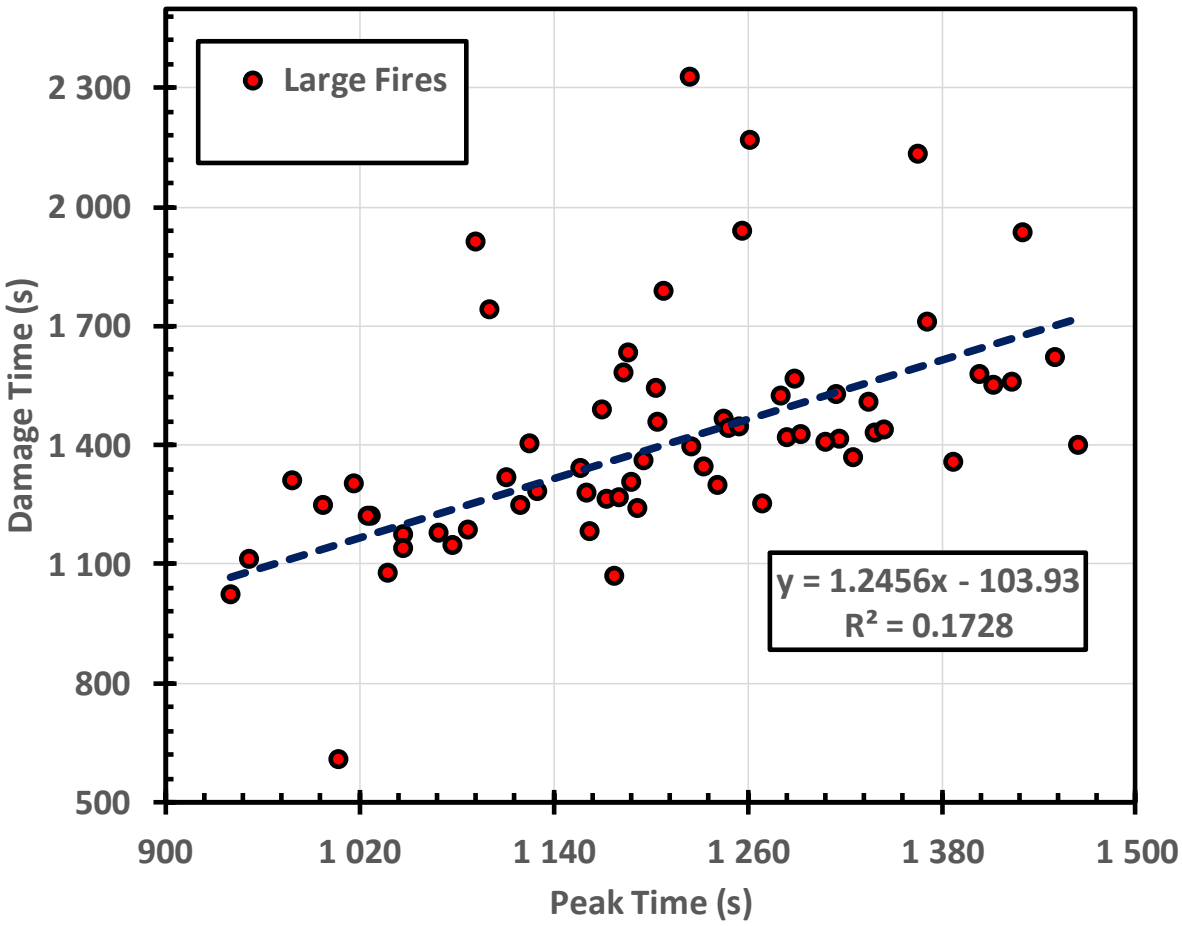


Figure 10. Damage time vs. peak time.

3. Simulation-based event trees

PRA software FinPSA (VTT 2019) includes a module for simulation-based event trees. The module has been developed for probabilistic analysis of severe nuclear reactor accidents (Tyrväinen et al. 2016, Tyrväinen & Karanta 2019), but it is, in practise, a general-purpose probabilistic risk analysis tool. The module combines event trees with computation scripts written using FinPSA's own programming language, containment event tree language (CETL). In the script files, user defines functions that calculate probabilities of event tree branches and possibly other variable values, such as amounts of consequences or timings of events. The script files enable use of various different modelling approaches, because contents of the scripts are not limited in any way, except that they must conform the CETL syntax.

The model includes a separate script file for each event tree section, for an initial section, and for a common section, which is common to all event trees in the project if there are multiple event trees. A function name is assigned to each event tree branch, and the function has to be defined in the script file of the corresponding event tree section. The function returns the probability of the event tree branch. It is also possible to write other functions that are called e.g. by branch functions. The model can include both global variables and local variables limited for a specific event tree section. Normal variable types, such as 'real', 'integer', 'Boolean' and 'string', can be used. Distributions of few different types can also be specified. A set of built-in functions is available, including some distribution operations.

To account for uncertainties related to variable values, it is possible to specify probability distributions for parameters and perform Monte Carlo simulations. At each simulation cycle, a value is sampled from each specified distribution, and based on that, numerical conditional probabilities are calculated for all event tree branches, and values are calculated for all variables at each end point of the event tree. After the simulations, statistical analyses are performed to calculate frequency and variable value distributions for each end point among other statistical results and correlation analyses. It is also possible just to calculate point values of the event tree based on the mean values of distributions. Event tree sequences can also be grouped by a binner routine, and combined results can be calculated for the specified consequence categories.

4. Model

There are several different ways how the case study can be implemented in FinPSA. Firstly, the fire simulation results could be imported to FinPSA as probability distributions of relevant variables or vectors containing all simulation data. The vectors are selected here, because this is the more accurate approach to utilise the simulation results. It can be done since the number of fire simulation cycles is quite small. If distributions of cable failure time and time of intolerable firefighting conditions were estimated based on the simulation data and implemented in FinPSA, it would be more difficult to take into account the correlation between those two variables (see Section 2.2). Using all simulation data, the correlation is naturally taken into account at the same level as it is present in the simulation data.

The second main modelling decision is related to the operation time model. It is possible to implement the entire operation time model in FinPSA or utilise distributions calculated with the Excel tool for simpler implementation of the FinPSA model. Even though both approaches could be equally accurate, it is decided that the entire model is re-implemented in FinPSA so that the new model is not dependent on the old Excel tool. The size of the model is not a problem, since it is significantly simpler than a typical level 2 PRA model.

The new FinPSA event tree is presented in Figure 11. The model is simplified so that it is assumed that cable damage is automatically prevented if the automatic fire suppression system works, which was a conclusion made in the previous analysis (Hostikka et al. 2012). The operation time model is divided between eight event tree sections according to the operation phases. Cable damage occurs in sequence 4. In sequence 3, cable damage is prevented by the fire brigade.

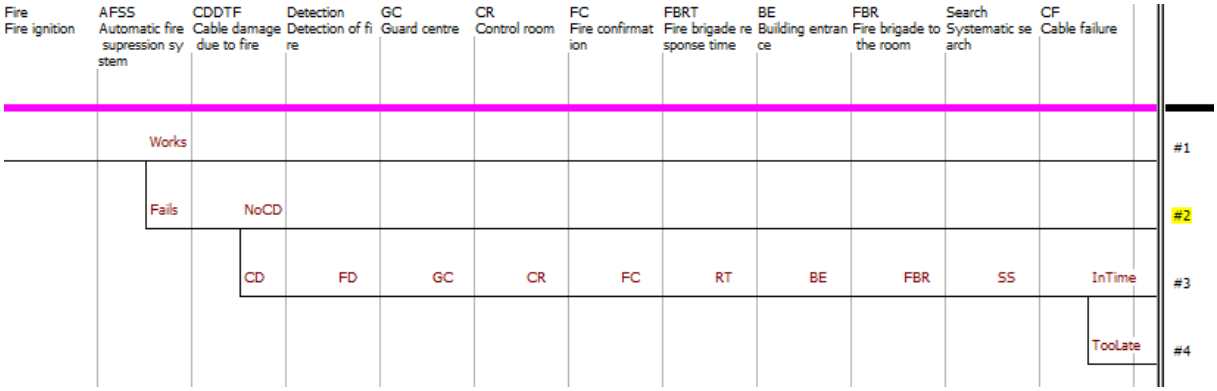


Figure 11: Event tree with operation time model.

As discussed earlier, fire scenarios are modelled in FinPSA by importing all relevant data as vectors. Relevant variables from fire simulations are the detection time, cable damage time

and time when the firefighting conditions become intolerable. For each variable, a vector of 66 elements is created, because cable damage occurred in 66 simulations according to temperature limit 180 °C. The simulations where cable damage did not occur are filtered out in event tree section Cable damage due to fire, i.e. the probability of the NoCD branch is set to 0.34. In FinPSA computation, for each of the 66 simulation cases, it is determined whether the cable failure occurs or not in every simulation cycle of FinPSA (operation time simulation cycle).

The scripts behind the first event tree sections are very simple. Probabilities of the automatic fire suppression system and cable damage due to fire are specified directly in the scripts. The initial section defines global variables and the vectors representing the fire simulation results.

The scripts related to the operation time related event tree sections are presented in the following. For each section, there is a routine 'init', where values are drawn for all random variables. There are three types of variables: probabilities of events (p-variables), time delay variables (t-variables) and random numbers between 0 and 1 used to determine whether an event occurs or not (r-variables). For each section, there is also a function of nil-type (which returns probability 1 in this case). These functions are used to calculate the values of time variables by summing the time delays depending on the events that occur.

Fire is primarily detected automatically by smoke detectors. If automatic detection fails, fire is detected by the senses, but it takes likely much longer time. Possibility that detection by the senses is faster than automatic detection is also taken into account in the model by comparing detection times. It is also possible that the fire is detected from a wrong room causing delay. The automatic detection times come from fire simulations (DetectTime vector), and for each fire simulation case, the real detection time is determined taking into account also the possibility of detection by senses or from a wrong room.

Detection of fire

```
real p_dar, p_adf, t_dar, t_md, r_adf, r_dar

integer i, n

routine init
  p_adf = raneven(0.001,0.02)  $ Automatic detection failure probability
  p_dar = raneven(0.001,0.002) $ Probability of detection from another room

  t_dar = raneven(1,15)        $ Detection from another room
  t_md = raneven(1,120)        $ Manual detection

  r_adf = random()
  r_dar = random()
return


function nil FD
  $ Detection time is determined for each fire simulation cycle
  i = 1
  $ While loop over the fire simulation cycles
  while lesse(i,NumSim) do
  begin
    $ Detection occurs automatically or manually
    if (r_adf > p_adf) and (DetectTime(i) < t_md) then
    begin
      $ Automatic detection time from fire simulation
      RealDetTime(i) = DetectTime(i)
    end
    else
    begin
      $ Manual detection
      RealDetTime(i) = t_md
    end

    $ Detection from another room?
    if r_dar < p_dar then RealDetTime(i) = RealDetTime(i)+t_dar
```

```
    i = i+1
  end

  $ One detection time is selected for the computation of operation time distribution
  n = simurun()
  while more(n,NumSim) do n = n-NumSim
  t_det = RealDetTime(n)
return nil
```

The guard centre has to make a collective alarm and call the unit director of fire brigade. If the guard centre personnel do not notice the automatic alarm, the guard centre becomes aware of the situation based on a call from the control room. Before that, the control room personnel have to set off the alarm. Wrong interpretation of the alarm by the control room can cause additional delay. Another additional delay occurs if the guard centre gives the fire brigade a wrong address.

Guard centre

```
real p_anf, p_wa, r_anf, r_wa, r_wi, t_mca, t_cud, t_wa

routine init
  p_anf = raneven(0.01,0.05) $ Auto-notification failure probability
  p_wa = raneven(0.01,0.05)  $ Probability of wrong address
  p_wi = raneven(0.01,0.05)  $ Probability of wrong interpretation of the alarm

  t_mca = raneven(0.1,0.5)   $ Making a collective alarm
  t_cud = raneven(1,1.5)     $ Calling the unit director
  t_wa = raneven(2,10)       $ Wrong address
  t_aso = raneven(0.5,5)     $ Alarm set-off
  t_wi = raneven(1,10)       $ Wrong interpretation of the alarm
  t_cgc = raneven(0.5,2)     $ Call to the guard centre

  r_anf = random()
  r_wa = random()
  r_wi = random()
return


function nil GC
  $ Guard centre operation time is determined
  t_gc = t_mca+t_cud

  if r_anf < p_anf then
  begin
    t_gc = t_gc+t_aso+t_cgc
    if r_wi < p_wi then t_gc = t_gc+t_wi
  end

  if r_wa < p_wa then t_gc = t_gc+t_wa
return nil
```

The control room has to set off the alarm, call to the guard centre and send a person to confirm the fire. Wrong interpretation of the alarm can cause additional delay. When the fire brigade has arrived to the fire location, the control room needs to collaborate with the fire brigade. A credibility gap can cause delay in the collaboration. The control room may also need to switch off voltage from the cables (commented out in the scripts below). Failure to switch off voltage from some cables can cause significant additional delay.

Control room

```
real p_cg, p_vno, t_sp, t_cg, t_sov, t_vno, r_wi, r_cg, r_vno

routine init
  p_cg = raneven(0.01,0.1)  $ Probability of credibility gap
  p_vno = raneven(0.03,0.3) $ Probability that voltage not switched off

  t_sp = raneven(0.5,2)     $ Sending a person to ensure fire
  t_co = raneven(1,10)      $ Collaboration
  t_cg = raneven(3,5)       $ Credibility gap
  t_sov = raneven(10,30)    $ Switching off the voltage
  t_vno = raneven(10,30)    $ Voltage not switched off
```

```
  r_wi = random()
  r_cg = random()
  r_vno = random()
return


function nil CR
  $ Control room operation time
  t_cr = t_aso+t_cgc+t_sp
  if r_wi < p_wi then t_cr = t_cr+t_wi

  $ Collaboration time
  if r_cg < p_cg then t_co = t_co+t_cg

  $ Delay from switching off voltage
  $ t_v = t_sov
  $ if r_vno < p_vno then t_v = t_v+t_vno
  t_v = 0
return nil
```

The control room has to send a person to confirm the fire. Walking to the room takes some time, and additional delay may be caused by selecting a wrong route.

Fire confirmation

```
real p_wr, t_msp, t_wro, t_wr, r

routine init
  p_wr = raneven(0.01,0.1) $ Probability of choosing a wrong route

  t_msp = raneven(0,15)    $ Moving to the starting point
  t_wro = raneven(2,3)     $ Walking to the room of origin
  t_wr = raneven(1,15)     $ Choosing a wrong route

  r = random()
return


function nil FC
  $ Fire confirmation time is determined and added to the control room operation time
  t_fc = t_msp+t_wro

  if r < p_wr then t_fc = t_fc+t_wr

  t_cr = t_cr+t_fc
return nil
```

For the fire brigade response time, a gamma distribution has been estimated based on statistics from drills carried out at the plant. Since, the programming language does not currently offer a gamma distribution function, the distribution is implemented as a DPD distribution of CETL. The DPD distribution is defined by 13 cumulative percentile values (0%, 2.5%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, 97.5%, 100%). The values between those values are calculated by linear interpolation by FinPSA. Additional delays to the response time may be caused by heading to a wrong target, crew or equipment being unavailable, route barrier and other alarm.

Fire brigade response time

```
real p_wt, p_uc, p_ue, p_rb, p_oa, t_wt, t_uc, t_ue, t_rb, t_oa,
     r_wt, r_uc, r_ue, r_rb, r_oa

DPD ResponseTime = (0.24,0.586,1.07,1.51,1.90,2.29,2.69,3.14,3.68,4.38,5.48,7.49,10)

routine init
  p_wt = raneven(0.02,0.2)    $ Probability of wrong target
  p_uc = raneven(0.05,0.5)    $ Unavailability of crew
  p_ue = raneven(0.01,0.1)    $ Unavailability of equipment
  p_rb = 0.06                 $ Probability of route barrier
  p_oa = 0.03                 $ Probability of other alarm

  t_rt = randpd(ResponseTime) $ Response time
```

```
  t_wt = raneven(2,4)          $ Wrong target
  t_uc = raneven(5,8)          $ Crew unavailable
  t_ue = raneven(5,10)         $ Equipment unavailable
  t_rb = raneven(5,10)         $ Route barrier
  t_oa = raneven(10,15)        $ Other alarm

  r_wt = random()
  r_uc = random()
  r_ue = random()
  r_rb = random()
  r_oa = random()
return


function nil RT
  $ Total response time is determined
  if p_wt > r_wt then t_rt = t_rt+t_wt
  if p_uc > r_uc then t_rt = t_rt+t_uc
  if p_ue > r_ue then t_rt = t_rt+t_ue
  if p_rb > r_rb then t_rt = t_rt+t_rb
  if p_oa > r_oa then t_rt = t_rt+t_oa
return nil
```

Time to the building entrance is the time from the fire truck to the door of the building. Additional delays may be caused by a broken hose, broken coupling or a pump failure.

Building entrance

```
real p_bh, p_bc, p_pf, t_bh, t_bc, t_pf, r_bh, r_bc, r_pf

routine init
  p_bh = raneven(0.02,0.1)  $ Probability of broken hose
  p_bc = raneven(0.01,0.05) $ Probability of broken coupling
  p_pf = raneven(0.01,0.05) $ Pump failure to start probability

  t_be = raneven(0.5,5)     $ Time to the building entrance
  t_bh = raneven(2,4)       $ Broken hose
  t_bc = raneven(2,4)       $ Broken coupling
  t_pf = raneven(2,4)       $ Pump fails to start

  r_bh = random()
  r_bc = random()
  r_pf = random()
return


function nil BE
  $ Total time to building entrance is determined
  if r_bh < p_bh then t_be = t_be+t_bh
  if r_bc < p_bc then t_be = t_be+t_bc
  if r_pf < p_pf then t_be = t_be+t_pf
return nil
```

The time it takes to move from the building door to the room where the fire is located has been calculated based on the distance and speed, which has been assumed constant. Additional delays may be caused by forgotten keys or wedges or a broken hose.

Fire brigade to the room

```
real p_fk, p_fw, p_bh, t_fk, t_fw, t_bh, r_fk, r_fw, r_bh

routine init
  p_fk = raneven(0.02,0.2) $ Probability that keys are forgotten
  p_fw = raneven(0.02,0.2) $ Probability that wedges are forgotten
  p_bh = raneven(0.05,0.3) $ Probability of broken hose

  t_re = 0.462             $ Time to the room entrance
  t_fk = raneven(3,6)      $ Forgotten the keys
  t_fw = raneven(1,2)      $ Forgotten the wedges
  t_bh = raneven(1,2)      $ Broken hose

  r_fk = random()
  r_fw = random()
  r_bh = random()
```

```
return


function nil FBR
  $ Total room entrance time is determined
  if r_fk < p_fk then t_re = t_re+t_fk
  if r_fw < p_fw then t_re = t_re+t_fw
  if r_bh < p_bh then t_re = t_re+t_bh
return nil
```

The fire brigade has to perform a systematic search to find the fire. Additional delays may be caused by a forgotten thermal camera, a loss of pressure from a hose or a communication problem.

Systematic search

```
real p_fc, p_pl, p_cp, t_fc, t_pl, t_cp, r_fc, r_pl, r_cp

routine init
  p_fc = raneven(0.02,0.2) $ Probability that the thermal camera is forgotten
  p_pl = raneven(0.05,0.3) $ Probability of pressure loss
  p_cp = raneven(0.05,0.3) $ Probability of communication problem

  t_ss = raneven(0,7)      $ Systematic search
  t_fc = raneven(0,3)      $ Forgotten the thermal camera
  t_pl = raneven(1,5)      $ Pressure loss
  t_cp = raneven(2,5)      $ Communication problem

  r_fc = random()
  r_pl = random()
  r_cp = random()
return


function nil SS
  $ Total search time is determined
  if r_fc < p_fc then t_ss = t_ss+t_fc
  if r_pl < p_pl then t_ss = t_ss+t_pl
  if r_cp < p_cp then t_ss = t_ss+t_cp
return nil
```

In the final section of the event tree, the operation time from the detection to the beginning of the systematic search is calculated. Then, for each fire simulation cycle, it is checked whether the cable failure occurs before the suppression, and whether the room conditions are such that the fire brigade can enter the room when they are ready. In other words, for each fire simulation, it is determined whether the fire brigade is in time or not. The probability of a cable failure is calculated based on the number of simulation cycles where the fire brigade was too late.

```
function real TooLate
  $ Operation time from the detection to the beginning of the systematic search is calculated
  t_oper = t_co+t_v
  if t_cr < t_gc+t_rt+t_be+t_re then
  begin
    t_oper = t_oper+t_gc+t_rt+t_be+t_re
  end
  else
  begin
    t_oper = t_oper+t_cr
  end

  i = 1
  c = 0
  $ While loop over the fire simulation cycles
  while lesse(i,NumSim) do
  begin
    if lesse(DamageTime(i),RealDetTime(i)+t_oper+t_ss) then
    begin
      $ Cable damage occurs before suppression
      c = c+1
    end
    else if lesse(BadConditions(i),RealDetTime(i)+t_oper) then
    begin
```

```
      $ Not possible to enter the room due to intolerable conditions
      c = c+1
    end
    i = i+1
  end

  $ Probability of cable failure
  p = c/NumSim

  $ Total operation time from the ignition
  t_oper = t_oper+t_det+t_ss
  Damage = true
return p
```

The FinPSA model was simulated 10000 cycles. The resulting cable failure probability was 0.0158 for the case where the voltage cut-off is not needed. The Excel-based PFS tool produced cable failure probability of 0.0154 with the equivalent operation time model (slightly different from (Hostikka et al. 2012)). Naturally, the simulation models produce slightly different results depending on the drawn random numbers. Both tools produced also approximately the same operation time distribution, and the average lengths of different operation phases were approximately the same. Based on this, it can be concluded that the new FinPSA model is capable of producing results sufficiently similar to PFS, even though the exactly same results cannot be reproduced.

The probability that the fire brigade suppresses the fire is quite small in this case, 0.00409. The sprinkler system is much more likely to prevent the cable failure. If the sprinkler system fails and the fire is large enough to damage the cable, the fire brigade is in time with a probability of 0.21.

Table 1 presents the average lengths of different operational phases. The correlation of the length of each operational phase to the cable failure probability is also presented. Based on the correlations, it can be identified that the control room operation, co-operation, systematic search and fire brigade response are to most critical phases with regard to a cable failure. These phases are also the phases that typically last longest. The best way to improve the result would be to improve the operation in these phases.

Table 1: Results of individual operational phases.

Variable	**Average (min)**	**Correlation to cable failure probability**
Detection time	1.88	0.057
Control room	15.9	0.45
Guard centre	1.86	0.074
Fire brigade response time	6.38	0.25
Fire brigade to the building entrance	3.13	0.082
Fire brigade to the room of origin	1.41	0.069
Co-operation	5.70	0.34
Systematic search	4.80	0.26

To study the impact of modelling the firefighting conditions, the model was changed so that the fire brigade was assumed to be able to enter the room in every case. The cable failure probability decreased only to 0.0150. This means that the contribution of cases where the conditions at the door prevent the firefighting even though the fire brigade is otherwise in time is quite small. The reason for this is that in most fire cases the conditions become intolerable after the cable failure or just before it. Regardless, the cable failure probability would be underestimated by not evaluating the firefighting conditions.

For future studies, the detection modelling can be simplified, because it was identified in Section 2.2 that the automatic detection time does not correlate significantly with the cable failure time. Instead of using explicitly the times from the fire simulation, automatic detection times can be drawn from a lognormal distribution estimated based on the fire simulation results. The detection of fire scripts can then be simplified as follows:

```
real p_dar, p_adf, t_dar, t_md, t_ad, r_adf, r_dar

integer i

routine init
  p_adf = raneven(0.001,0.02)  $ Automatic detection failure probability
  p_dar = raneven(0.001,0.002) $ Probability of detection from another room

  t_dar = raneven(1,15)        $ Detection from another room
  t_md = raneven(1,120)        $ Manual detection
  t_ad = ranlogn(1.23,2.46)    $ Automatic detection time

  r_adf = random()
  r_dar = random()
return


function nil FD
  $ Detection time is determined
  if (r_adf > p_adf) and (t_ad < t_md) then
  begin
    t_det = t_ad
  end
  else
  begin
    t_det = t_md
  end

  if r_dar < p_dar then t_det = t_det+t_dar
return nil
```

Some minor changes in the cable failure section are also needed. This new model produces results very close to the model presented earlier.

5. Discussion

In this study, a fire PRA model including an operation time model of firefighting actions was implemented in a tool that has not been used for this purpose previously, FinPSA. Compared with the Excel tool PFS, the FinPSA model of operation times seems better structured, and easier to read and maintain. FinPSA is easier to use than the heavy Excel file, though the user must first learn to use FinPSA and learn the CETL syntax. It is easier to write complex computation in FinPSA scripts than in Excel. Excel provides possibilities to customize the presentation of results, whereas FinPSA does not, but FinPSA simulation data can also be exported to Excel. Excel suits better for handling of large amounts of input data, though in this case, the size of the data is quite limited. A benefit of PFS is also that it creates the fire simulation cases automatically and manages the simulations. In that purpose, it is not practical to replace PFS by FinPSA.

It was easy to import all relevant fire simulation data in FinPSA, because the number of simulation cycles was so small. However, with larger number of simulation cycles, this might

not be practical. The current limit for a vector in FinPSA is 1000 elements, but already for hundreds of simulation cycles, it would become tedious to import the data manually. One possibility would be to develop an automatic import routine that would read the data from an input file. Alternatively, the modelling approach could be changed. Fire simulations could be divided into suitable categories so that modelling of dependencies could be done sufficiently accurately inside those categories without importing all the data (e.g. by importing estimated distributions of relevant variables taking into account the dependencies). The fire categories could have separate event tree branches. For example, simulations with late cable failure (e.g. later than 30 minutes) could be one category according to Figure 4, because the correlation between the cable failure time and firefighting conditions is different in those cases compared to other cases. Anyhow, Monte Carlo fire simulations by a CFD-simulator are so heavy that it is unlikely that the number of realizations will increase very much in the near future.

One option for further development of the modelling approach would be to utilise event tree based modelling more, i.e. to create branches for events related to the fire brigade operations. The benefit of the branching would be to put more focus on rare events. For example, the case of having a credibility gap could be evaluated this way in every simulation cycle instead of about 5.5% of the simulation cycles. This would increase the accuracy of the analysis. However, the model contains so many random events that the event tree would become too large if branches were created for all events. A more realistic option would be to create branches only for the most important events, which would have to be identified first. This particular analysis case is however such that it is more likely that the fire brigade is late than in time. This means that there is not much need to put more focus on delay events, since it is uncertain whether the fire brigade is in time even without the delays, and the fire brigade being in time is actually the rarer event. With this regard, the analysis case differs significantly from normal NPP PRA. The branching approach would be more useful if the case was such that the fire brigade being late was a rare event. Such fire analysis cases exist too, e.g. in (Kloos et al. 2014), so the approach is worth considering in future studies.

Another potential area for future development is uncertainty analysis, i.e. computation of an uncertainty distribution for the cable failure probability. The model already contains uncertainty distributions for various time delays and uncertainties in fire progression are also taken into account, but proper uncertainty analysis of the cable failure probability would require separation of aleatory and epistemic uncertainties (Durka Rao et al. 2007), which are now totally mixed in the model. The reason for this is that the probability of the cable failure itself represents the aleatory uncertainty on the occurrence of the cable failure, and the uncertainty distribution of the probability should represent uncertainty due to lack of knowledge, i.e. epistemic uncertainty. To perform the analysis, for each time variable, it should be analysed what is the uncertainty caused by lack of knowledge (epistemic) and what is the uncertainty caused by the randomness of the action or event (aleatory). It would mean that for each time variable, there would be a distribution representing aleatory uncertainty, and there would also be distributions for the parameters of that distribution representing epistemic uncertainty. Similarly, epistemic and aleatory uncertainties should be separated in fire progression modelling. Boneham et al. (2019) have developed a systematic process for the identification and characterization of uncertainties in fire PRA. The uncertainty analysis sets also requirements for the computation tool. Epistemic and aleatory variables could e.g. be handled in separate sampling loops as presented in (Tyrväinen & Karanta 2019) and (Boneham et al. 2019). This type of functionality has however not been implemented in FinPSA yet.

6. Conclusions

In this study, probabilistic analysis of an old nuclear power plant cable room fire scenario has been successfully implemented in a new tool, simulation-based event tree of FinPSA. The main components of the model are Monte Carlo fire simulations and a stochastic operation

time model for firefighting. The fire simulations are performed separately using FDS software, and the results of the simulations are imported to FinPSA. The operation time model is implemented in FinPSA scripts in eight parts corresponding to different operational phases. The results calculated using FinPSA are approximately same as the results calculated by Excel-based Probabilistic Fire Simulator (PFS). FinPSA offers better model structure, better readability and better maintainability than PFS. In the script files of FinPSA, it is easier to develop complex computation rules. FinPSA may therefore be considered a useful and practical tool for fire PRA modelling.

The study could be continued e.g. by modelling firefighting actions and their dependency on fire progression more realistically. Cooling from the door could also be an option for the fire brigade. Other conditions than visibility at the door could be taken into account, and fire brigade could possibly enter the room from a different door if it is not possible from the first door. The cable room fire scenario should also be updated for present situation, including more accurate analysis of the automatic suppression system and the new guidance for the voltage cut off procedure. The study could also be extended by analysing more consequences, e.g. failures of different cables and fire spreading. This would require new fire simulations. Larger number of simulation results would increase the accuracy of the analysis, but could also introduce new challenges for the modelling. The long-term goal is to develop a general-purpose approach for fire PRA. It however requires analysis of a wider range of fire scenarios, because different modelling techniques may be needed in other fire analysis cases. Another possibility for improvement would be to perform comprehensive uncertainty analysis.

References

Boneham, P., Georgiev, G., Guymer, P. (2019). An approach to fire probabilistic risk assessment modeling, uncertainty quantification and sensitivity analysis. PSA 2019, Charleston, SC, 28 April - 3 May, 2019.

Durga Rao, K., Kushwaha, H.S., Verma, A.K., Srividya, A. (2007). Quantification of epistemic and aleatory uncertainties in level-1 probabilistic safety assessment studies. Reliability Engineering and System Safety, 92, 947-956, 7.

Hostikka, S. (2008). Development of fire simulation models for radioactive heat transfer and probabilistic risk assessment [doctoral dissertation], VTT publication 683. VTT Technical Research Centre of Finland Ltd, Espoo.

Hostikka, S., Keski-Rahkonen, O. (2003). Probabilistic simulation of fire scenarios. Nuclear Engineering and Design, 224, 301-311.

Hostikka, S., Kling, T., Paajanen, A. (2012). Simulation of fire behaviour and human operations using a new stochastic operation time model. 11th International probabilistic safety assessment and management conference & the annual European safety and reliability conference (PSAM11 & ESREL 2012), Helsinki, 25-29 June, 2012.

Kling, T. (2010). Palo-HRA - Operatiivisen toiminnan malli ydinvoimalaitosten paloriskien todennäköisyyspohjaiseen arviointiin, Versio 2, VTT-R-07211-10. VTT Technical Research Centre of Finland Ltd, Espoo.

Kloos, M., Peschke, J., Forell, B. (2014). Insights from and Integrated Deterministic Probabilistic Safety Analysis (IDPSA) of a Fire Scenario. Probabilistic Safety Assessment and Management PSAM 12, Honolulu, Hawaii, June, 2014.

McGrattan, K, McDermott, R, Weinschenk, C, Overholt, K, Hostikka, S, Floyd, J. (2013). Fire Dynamics Simulator User's Guide, Sixth edition. National Institute of Standards and Technology, Gaithersburg, MD, NIST special publication 1019.

OECD (2015). International workshop on fire probabilistic risk assessment (PRA), NEA/CSNI/R(2015)12. OECD Nuclear Energy Agency, Paris.

Sakurahara, T., Reihani, S.A., Mohaghegh, Z., Brandyberry, M., Kee, E., Johnson, D., Rodgers, S., Billings, A. (2014). Developing a New Fire PRA Framework by Integrating Probabilistic Risk Assessment with a Fire Simulation Module. Probabilistic Safety Assessment and Management PSAM 12, Honolulu, Hawaii, June, 2014.

Silvonen, T. (2013). Steam explosion case study using IDPSA methodology, VTT-R-05974-13. VTT Technical Research Centre of Finland Ltd, Espoo.

Siu, N., Sancaktar, S. (2015). Fire PRA Maturity and Realism: A Technical Evaluation and Questions, International Workshop on Fire Probabilistic Risk Assessment (PRA), NEA/CSNI/R (2015)12. OECD Nuclear Energy Agency, Paris.

Tyrväinen, T., Karanta, I. (2019). Dynamic containment event tree modelling techniques and uncertainty analysis, VTT-R-06892-18. VTT Technical Research Centre of Finland Ltd, Espoo.

Tyrväinen, T., Silvonen, T., Mätäsniemi, T. (2016). Computing source terms with dynamic containment event trees. 13th International conference on probabilistic safety assessment and management (PSAM13), Seoul, Korea, 2-7 October, 2016.

VTT Technical Research Centre of Finland Ltd (2019). FinPSA – Tool for promoting safety and reliability. https://www.simulationstore.com/finpsa (link accessed 24.9.2019)