SCHEMA-SI : A HYBRID FIRE SAFETY ENGINEERING TOOL

A. Muller[3] (formerly 1-2), F. Demouge[1], M. Jeguirim[2], Ph. Fromy[1], and J.-F. Brilhac[2]
anne.muller@utt.fr
1: Université Paris–Est, Centre Scientifique et Technique du Bâtiment, Marne-la-Vallée, France
2: Université de Haute Alsace, Laboratoire Gestion des Risques et Environnement, Mulhouse, France
3: Université de Technologie de Troyes, UMR STMR CNRS, Institut Charles Delaunay, Troyes, France

Abstract

SCHEMA-SI is an engineering tool able to evaluate the performance of building fire safety system. It uses a dynamic hybrid model, which means that discrete events and continuous phenomenon are interconnected at each instant. The model has been developed as a tool to assist fire protection engineers in performing fire safety engineering calculations. This tool may be also used to conduct risk assessments and to evaluate whether selected design strategies are sufficiently safe in case of fire in a specific building. This paper describes the basis of the SCHEMA-SI tool. Sub-models used to perform calculations are discussed.

1 Introduction

Fire Safety Engineering (FSE) aims at designing building fire safety measures. To achieve that goal, it relies on numerical simulation in order to predict specific fire scenarios. Scenarios analysis provides an estimation of the risk in case of fire for a specific building configuration. Various numerical tools have been proposed to predict fire scenarios and estimate fire risk [1]. Some of them, like FDS (Fire Dynamic Simulator) are commercially available and have been largely employed by fire safety engineer. Most of these tools match deterministic model and are often dedicated to one aspect of the fire. For example, CFAST and FDS [2] are dedicated to fire and smoke spread, SAFIR [3] concerns thermo-mechanical behaviour of building structures, EXODUS [4] deals with human evacuation and DETACT [5] predicts detector responses. Unfortunately, these tools are not adapted to predict real complex fire scenarios. In fact, such scenarios implicate combined effects between occupant behaviour, smoke spread, safety system or failure per example. Hence, several fire risk assessment models have been developed [6] [7]. These models incorporate probabilistic modelling techniques and often combine different sub-models (e.g.: fire and smoke spread, detection, alarm, occupant egress, etc.). Among these models, FIRECAM [9], FIERAsystem [10], HAZARD [11] and Fire Probabilistic Simulator (FPS) [12] are currently used in performance-based design. These tools are based on the sequential use of sub-models that exchange data automatically or manually. However, these tools do not allow sub-model interconnections continuously over time. For this reason, the development of a new computational tool for fire risk assessment was undertaken at the CSTB (French Scientific and technic Building Centre). This tool is called SCHEMA-SI (Stochastic Computation and Hybrid Event Modelling Approach – Sécurité Incendie). This paper discusses the basis of the SCHEMA-SI tool and starts by a brief introduction of the tool features (see section 2). After this introduction, main sub-models are explained (see sections 3 and 5). At last, the way sub-models are interconnected is defined (see section 5).

2 Main features of the SCHEMA-SI tool

The tool principles are summarised in Figure 1. This tool relies on Monte Carlo method [13] [12] in order to generate thousands of fire scenarios from one stochastic framework. As a result, these thousands of fire scenarios are used to **assess fire risk** in a specific building configuration. The stochastic framework contains a scope of potential behaviours (e.g.: human behaviour and lethality, fire time-evolution, safety system responses and dysfunction, etc.) described by a set of probabilistic parameters (e.g.: evacuation time, heat release rate, detector lapse rate, etc.). Random processes are introduced to overcome some uncertainties concerning input data and to describe better the variability of possible situations.

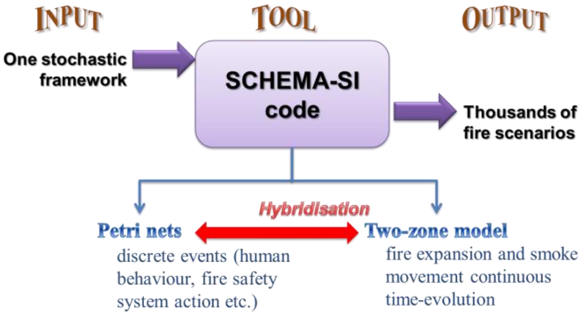


Figure 1 : SCHEMA-SI's principles

Technically, SCHEMA-SI is based on the hybridisation of two sub-models

On one hand, it relies on fire expansion and smoke movement continuous time-evolution modelling and calculation. Quantities such as gas and wall temperature or smoke layer height are computed by integrating a traditional system of equations of a **two-zone sub-model.** The two-zone model implemented in SCHEMA-SI is CIFI2009 (see 3).

On the other hand, SCHEMA-SI relies on modelling **discrete events** mainly representing human behaviour and fire safety system functioning (including failure). For example, smoke detector activation and door opening are represented by the way of discrete events. Events modelling use a specific **Petri net** formalism (see 4).

As a result of the **hybridisation** (see 5) SCHEMA-SI is able to model **interactions** between continuous phenomena (e.g.: fire evolution) and discrete phenomena (e.g.: door opening). Indeed, the realisation of certain events induce modifications in the equations system (e.g. if a person opens a door, mass and energy flow through the door are taken into account in mass and energy balances). In addition, quantities calculated by CIFI2009 may initiate discrete event (e.g.: when the upper layer temperature rise above a specific limit, people in the room die).

3 CIFI 2009 sub-model

CIFI 2009 predicts the transport of heat and smoke in the premises. The model was developed by the CSTB and was already validated for different configurations such as room fire [14]. The model matches a multi-compartment two-zone model. Multi-compartment model means many interconnected compartments may be taken into account simultaneously. Zone model are characterised by the modesty of calculation resources required, making their use consistent with Monte Carlo simulations. In two-zone models, the space of a compartment is divided into two gas zones in which the physical quantities (e.g.: temperature, chemical species composition, opacity…) are uniform and unsteady. The idea of dividing space into two zones comes from the observation that the hot gases from a fireplace accumulate under the ceiling, leaving space below free for fresh air. This applies to the premises of classical geometry (i.e.: parallelepipeds) and for fire pits placed in the lower part of the room, quite powerful relative to the size of local.

In CIFI 2009, a compartment room is divided into two control volumes: a relatively hot upper layer z_h and a relatively cool lower layer z_b (see Figure 2). Layers are separated by a horizontal virtual surface traversed by mass and energy fluxes dues to fire pits. The height of this interface, called the "thermal interface height" and noted Z varies over time. Figure 2 illustrates zone decomposition and summarizes the nature of mass and energy fluxes exchanged between these zones. This figure corresponds to a situation where fresh air arrives directly from the doorway into the lower layer and where hot gases exhaust out of the room from the same doorway. CIFI 2009 solves the mass and energy balance for each zone in order to predict the evolution of physical quantities over time. Equations involves are not detailed in this paper as zone model are relatively well-known for decades.

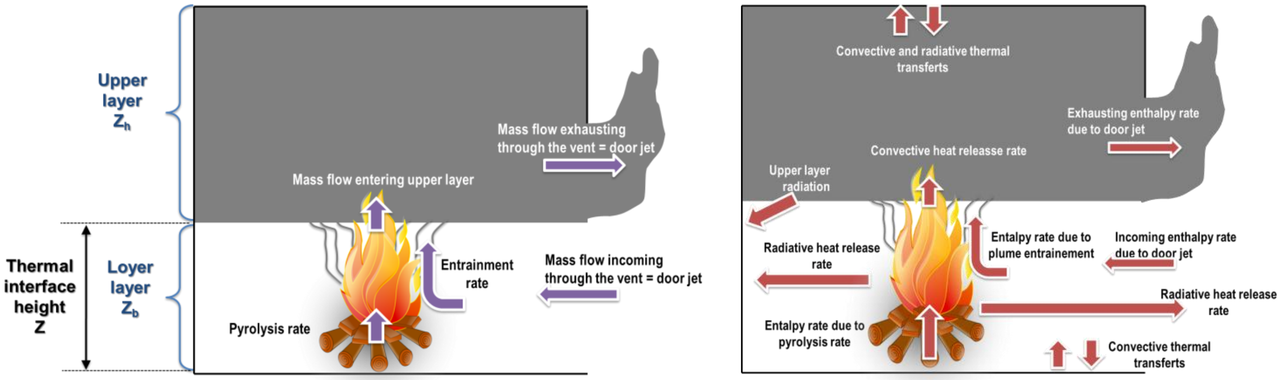


Figure 2 : Main mass (left) and energy (right) flows for one single burning object in an vented compartment

Numerically, CIFI 2009 is decomposed into a **main routine** calling **several sub-routines**. The **main routine** contains the **differential equation systems solver** and the **radiation model**. **Sub-routines** are dedicated to **sources terms** (e.g.: combustion model), other **heat transfers** (e.g.: conduction model) as well as to **net mass and energy fluxes calculation** (e.g.: gas flow through vents). It is very important to notice this decomposition in sub-routines as it is used to perform the hybridisation with Petri nets. Another numerical trick to notice is that gas characteristics (e.g.: upper layer temperature) are not computed at the same time than the solid characteristics (e.g.: temperatures gradient inside walls). This trick permits to save calculation time. The loss of accuracy is neglected because solid thermal evolution is slower than the gas one.

Next section is dedicated to the Petri nets formalism.

4 Petri nets sub-model

The Petri nets formalism used in SCHEMA SI tools belongs to the high-level Petri nets [15] [16]. In particular, this formalism is called Object-Oriented Differential Predicate-Transition (OO-DPT) Petri net [17]. This typical formalism has significant differences from the original formalism [18]. The main features of the OO-DPT Petri net formalism are listed in subsection (see 4.1). The original formalism is detailed in literature. This paper proposes some basic mathematical definitions and illustrations (see 4.2) as it is relatively new and little-known.

4.1 Features of the OO-DPT Petri net formalism

The OO-DPT Petri net formalism has mainly three features discussed below.

Firstly, the OO-DPT Petri net formalism allows modelling hybrid dynamic system. The classification of a system as “hybrid” concerns the nature of the variables used during building system models. In this sense, for modelling purposes, systems could be classified as Discrete Event Dynamic Systems (DEDS) when state variables are represented by integer numbers or logic variables; or as Continuous Variables Dynamic Systems (CVDS) when state variables can be represented by real numbers [19]. Hybrid systems mix the characteristics of DEDS and CVDS including both discrete and continuous variables. This combination of the discrete and continuous variables is essential for integrating the physical differential equations of CIFI 2009 in Petri nets. In the SCHEMA-SI tool, CVDS modelling corresponds to fire spread and smoke movement while DEDS modelling corresponds to discrete events such as safety system response, occupant behaviour, building component response, ignition and flashover and extinction occurrences.

Secondly, the OO-DPT Petri net formalism allows introducing randomised parameters. This feature is used in SCHEMA-SI in order to take into account unpredictable aspects of fire accidents. For example, the SCHEMA-SI tool takes the following randomised aspects:

- **initial conditions** (e.g. : vents initial state, people initial location, number of people inside the premises, fire location, initial burning item…) ;
- **event causes** (e.g. : conjunction, conditionality, occurrence frequency, time or sensitivity intervals…) ;
- **event consequences** (e.g. : event success or failure…).

The last feature of the OO-DPT petri net is object-oriented (OO) modelling. According to the OO paradigm, the model of a system is composed of a set of interconnected objects. An object represents a physical entity such as a device, a person or a sensor. In the formalism, each object is represented by a sub-Petri net which models its behaviour. The marking of the sub-net (see 4.2.1 for additional explanations) indicates the current state of the object. In addition, the formalism tackles interconnections between objects via fusion of transitions (discrete interconnection) and variables sharing (continuous interconnection). Interconnections are more detailed in next section.

4.2 Mathematical description of the OO-DPT Petri net formalism and illustration on a simple example

Each k’st OO-DPT **sub-net** is composed of a structure, marking and annotations. These concepts are explained in details using two very simple sub-nets shown in Figure 3. These sub-nets correspond to a heat detector (sub net n°1) alerting the safety officer (sub net n°2). This example is considerably simpler than sub-nets tackled by SCHEMA-SI for a real case study but is adequate for explanations.

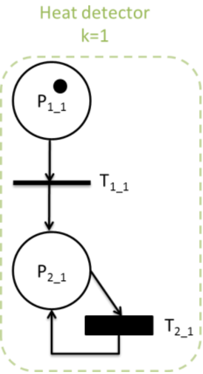
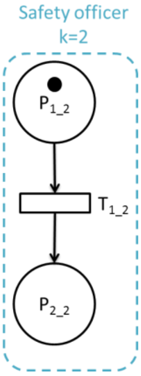
Structure and marking	**Annotations** *Nota : nomenclature is provided below – see 4.2.2*	
	**Variables :** $X_{ci\_1}: T_d; \propto$ $X_{vi\_1}: Po$ $X_{cp\_1}: \dots$ $X_{vp\_1}: T_g$ $X_{im\_1}: \dots$ **Enabling functions :** $e_{1\_1}: \dots$ $e_{2\_1}: (T_g > T_d) \wedge Po$	**Junction functions :** $j_{1\_1}: Po = U(1; 100) \leq 99$ $j_{2\_1}: \dots$ **Equation systems :** $F_{1\_1}: \dots$ $F_{2\_1}: T_g = \propto . t^2$ **Information for fusions of transitions :** T_{2_1} => T_{1_2} – alerts safety officer
	**Variables :** $X_{ci\_2}: \dots$ $X_{vi\_2}: \dots$ $X_{cp\_2}: \dots$ $X_{vp\_2}: \dots$ $X_{im\_2}: \dots$ **Enabling functions :** $e_{1\_2}: \dots$	**Junction functions :** $j_{1\_2}: \dots$ **Equation systems :** $F_{1\_2}: \dots$ $F_{2\_2}: \dots$ **Information for fusions of transitions :** …

Figure 3 : Example of two interconnected OO-DPT sub-nets

Structure and marking appear in the left column and annotations appear in the right column. The structure and marking are discussed first before providing the annotations significance in a second time.

4.2.1 Structure and marking

The sub-Petri nets **structure** shown in Figure 3 consists of four places (see white circles on Figure 3), three transitions (see rectangles and line on Figure 3) and six arcs (see arrows). **Places** symbolise objects states as follows:

- P_{1_1} : heat detector is not initialised yet ;
- P_{2_1} : temperature rises near the heat detector ;
- P_{1_2} : safety officer is not alerted yet ;
- P_{2_2} : safety officer is aware of a fire in the building.

Transitions symbolise events as follows:

- T_{1_1} : initialisation transition (used to initialise random variables) ;
- T_{2_1} : detection occurs and alarm rings ;
- T_{1_1}: safety officer hears the alarm.

Three kinds of transitions are distinguished in the OO-DPT formalism:

- internal transition (see T_{1_1}) : the event only concerns the object represented by the sub-net. Such transition are symbolised by lines ;
- active transition (see T_{2_1}) : the event has consequences on other objects (e.g. : when the alarm rings, safety officer is alerted). Such transition are symbolised by black rectangles ;
- passive transition (see T_{3_1}) : the event is caused by another object (e.g.: safety officer alert is initiated by the alarm ring).

Nota: active and passive transitions are used to tackle interconnections between sub-nets.
Arcs link events (transitions) and states (places).

The sub-Petri nets **marking** (see tokens – black dots) indicated object state. In Figure 3, marking corresponds to initial marking and indicates that:

- heat detector is not initialised yet (token in P_{1_1});
- safety officer is not alerted yet (token in P_{1_2}).

Marking evolution is implicated by transitions firing and matrix computation. Conditions for transitions firing obey to traditional Petri nets laws (input places must contain enough tokens) plus some laws specific to the OO-DPT formalism. These laws are detailed below.

4.2.2 Annotations

In order to model the object dynamics, the following elements are associated with the Petri net of each sub-net:

- a set of variables ;
- a set of enabling functions ;
- a set of junction functions ;
- a set of equation systems ;
- a set of information for fusion of transitions.

The **variables**, designated X, are used to characterise the system and its behaviour. Four variables are presented in the example shown in Figure 3:

- T_g : time-dependent gas temperature in the room where the heat detector is located;
- $\propto$: constant coefficient representing the combustion kinetics in the room;
- T_d : constant temperature detection level at which the heat detector is activated;
- Po : Boolean variable: « True » for detector availability, «false» for detector failure.

The variables of the k'st sub-net are divided into the following **five kinds of variables**:

- constant and internal variables X_{ci_k} : the value is constant for every scenarios and can only be read by the object itself (e.g. : $\propto$ and T_d) ;
- mobile and internal variables X_{vi_k}: the value change during a scenario or from one scenario to another. In addition, the value can only be read by the object itself (e.g. : Po whose value is not the same in all scenarios because the detector is not always available) ;
- constant and public variables X_{cp_k} : the value is constant for every scenarios and can be read be other objects;
- mobile and public variables X_{vp_k} : the value change during a scenario or from one scenario to another. In addition, the value can be read be other objects (e.g. : T_g whose value may be useful to other object in the same room) ;
- image variables X_{im_k} is a copy of a public variable value from another object.

Public and image variables are used to allow **continuous communication between sub-nets**. It corresponds to variables sharing.

Enabling functions, designated e, are Boolean expression implicating variables. They aim at adding conditions that must be respected to fire transitions. For this reason, each enabling function e_i is associated to a specific transition j. As a consequence, there are always as many enabling conditions as transitions. In the formalism, a transition is enabled only if the marking is suitable for firing (that is to say that input places contain enough tokens) and the corresponding enabling function returns the “True” value. In Figure 3, e_{1_1}: ... means no enabling function is added to T_{1_1}. As a consequence, T_{1_1} is fired as soon as P_{1_1} contains a token. However, enabling condition e_{2_1} is not blank. This means that T_{2_1} is fire as soon as P1_1 contains a token **and both the following conditions are respected:**

- the gaz temperature T_g exceeds the detection temperature T_d (cf. $T_g > T_d$) ;
- AND (cf. $\wedge$) detection system is available (cf. Po).

Junction functions, designated j, are mathematical expressions used to change variable values. As the change occurs with a transition firing, junction functions are used for **discrete modifications**. Each junction function j_j is associated to a specific transition j. As a consequence, there are always as many junction functions as transitions. In Figure 3, $j_{1_1}$is used to set randomly the value of Po by using a random function designated U (see footnote 1). The value is “True” in 99% of the scenarios and “false” in the other 1%. That is to say the detection system availability rate is 99% in this example. Other junction functions on are blank (see j_{2_1} and j_{1_2}).

Equation systems, designated F, are also mathematical expressions used to change variable values. However, as the change occurs when a place is marked, equation systems are used for **continuous modifications**. Each equation system F_i is associated to a specific place i. As a consequence, there are always as many equation systems as place. In Figure 3, F_{2_1} is used to increase T_g over time when place P_{2_1} contains token.

Information for fusions of transitions contains information required to merge transitions from different sub-Petri nets. Fusion of transition is used to allow **discrete communication between sub-Petri nets**. Fusion of transitions implicated that designated actives transitions (such as $T_{\_1}$ in Figure 3) will merge with designation passive transitions (such as T_{1_2} in Figure 3). Association designation is performed in information for fusions of transitions. Basically, consequence of fusion of transition is that two (or more) transitions in different sub-nets are fired at the same time. Firing only happens when all transitions implicated in the concerning fusion are enabled.

5 Building SCHEMA-SI by hybridising Petri nets and CIFI 2009

The basic structure of SCHEMA SI is a two-layer zone model for multiple rooms (CIFI 2009) coupled the OO-DPT Petri net formalism. Main features for the hybridisation are presented in the following sections.

5.1 Variable sharing between CIFI 2009 and Petri nets

Some variables computed by CIFI 2009 are characterised by a very general scope and shall be known in the whole environment SCHEMA-SI (e.g.: upper and lower layer temperatures, thermal interface height, molar fraction of carbon monoxide in both upper and lower layers…). As a consequence, these variables may be used in any sub-Petri nets to build junction functions, enabling functions and equations systems. However, chosen variable must firstly be defined as image variables in the subnet for this share to be valid. An example of variable sharing between CIFI 2009 and a sub-Petri net is provided in section (5.3).

---

[1] Function U(x_1;x_2) returns a relative number between x_1 and x_2 with a uniform distribution law.

5.2 Calling CIFI 2009 sub-routines in Petri nets

In the SCHEMA-SI tool, sub-Petri nets are used to model the behaviour of any objects implicated in the fire accident, except gas volume inside the premises. Indeed, phenomena related to gas volume (e.g.: smoke filling and transport) are fully supported by CIFI 2009. Some specific objects (e.g.: vents, fire sources and plumes, walls…) exchange matter and/or energy with gas zones. As explained in section 3, mass and energy flows between objects and gas zones are predicted by CIFI 2009 sub-routine calls. In addition, 1-d heat conduction problem is also solved in by CIFI 2009 sub-routine calls. In order to achieve hybridisation, **sub-routines calls are integrated into sub-Petri nets.** Particularly, there are two ways to call CIFI 2009 sub-routines in sub-Petri-nets:

- either in an **equation system** ;
- or in a **junction function.**

In further terms these calls are carried:

- either during the **marking of a place** (for equation system) ;
- or during **firing of a transition** (for junction functions).

Choosing the type of call depends on what is computed by the called sub-routine:

- when **predicting mass or energy flows** between the object and the gas, the call is necessarily done in an equation system ;
- when **solving 1-d heat conduction problem for solid materials**, the call is necessarily done in a junction functions. The concerned transitions must then be fired periodically (e.g.: every second).

The reason of this difference is the decoupling of gas characteristics calculation and conduction problem solving in the CIFI 2009 model.

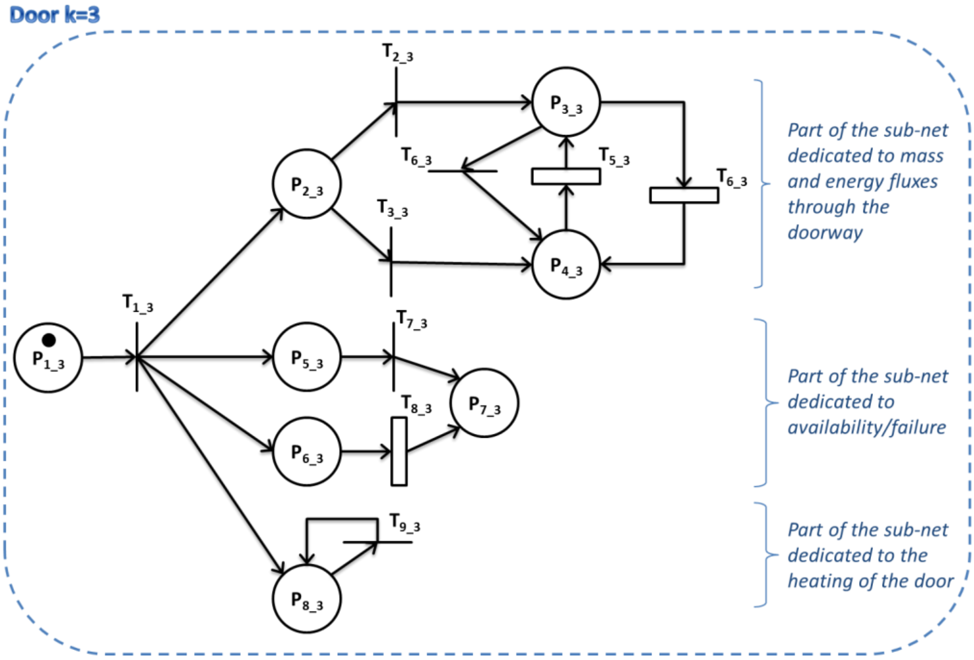
As illustrated in the following example, each CIFI 2009 sub-routine is referred by a specific letter in sub-Petri nets. Hence, from a Petri nets point of view, CIFI 2009 sub-routines are considered as black boxes that transform input variables X_i into output variables $Y_{i.}$

5.3 Example of an hybrid sub-Petri nets compatible with SCHEMA-SI

The sub-Petri net described in Figure 4 represents an example of common door. The behaviour of this specific door is the following. It has two potential opening states: fully closed or fully open. This state varies over time. In addition, the initial state varies from a scenario to another. During any scenario, this door may be destroyed either by heat or by another object (e.g.: by fire-fighters). The corresponding Petri net is a bit more complicated than those of Figure 3 but fits the kind of Petri nets involved in real fire configuration.

The sub-net is composed of eight places to which match eight equations systems. It also contains nine transitions to which match nine enabling functions and nine junction functions. In addition, the Petri nets structure is decomposed in three parts: the first indicated the opening states and is dedicated to the calculation of mass and energy fluxes thought the doorway; the second indicates if the door is available or not for handling; the third predicts heat transfers between the door itself and surrounding gas. This last part permits to calculate the total energy accumulated in the door and then to predict its failure. The sub-Petri net is fully explained below the figure.

Structure and marking



where :

P_{1_3} : first initialisation place – variables are not initialised yet
P_{2_3} : second initial place – variables are initialised but the initial opening state is not initialised yet
P_{3_3} : the door is closed – no mass or energy flux through the doorway
P_{4_3} : the door is open – mass and energy fluxes through the doorway are computed
P_{5_3} : the door is undamaged
P_{6_3} : the door is undamaged
P_{7_3} : the door is totally damaged
P_{8_3} : the doors receives an incoming energy flux from the surrounding gas, it is under stress

T_{1_3} : initialisation transition used to fire random variables
T_{2_3} : transition used to initialize the initial opening state of the door – fired if the door is initially closed
T_{3_3} : transition used to initialize the initial opening state of the door – fired if the door is initially open
T_{4_3} : door opening by another object (e.g. : by an person)
T_{5_3} : door closing by another object (e.g. : by a person or a door-closer)
T_{6_3} : door opening after its downfall
T_{7_3} : thermal destruction of the door
T_{8_3} : mechanical destruction of the door (e.g. : by fire-fighters)
T_{9_3} : calculation of the total energy accumulated in the door since the beginning of the scenario

Annotations

Variables :
$X_{ci\_3}: \{H; W\}$
$X_{vi\_3}: \{E_0; e_p; e_{p,s}; t_{maj}; blo_1\}$
$X_{cp\_3}: \{ip_1; ip_2\}$
$X_{vp\_3}: \ldots$
$X_{im\_3}: \{T_{zh}; T_{zb}; \Delta P; Yo_{2,zh}; Yo_{2,zb}; Y_{f,zh}; Y_{f,zb}; Z\}$

Enabling functions :
$e_{1\_3}: \ldots$
$e_{2\_3}: E_0 = 1$
$e_{3\_3}: E_0 = 2$
$e_{4\_3}: \overline{blo_1}$
$e_{5\_3}: \overline{blo_1}$
$e_{6\_3}: blo_1$
$e_{7\_3}: e_p \geq e_{p,s} \wedge \overline{blo_1}$
$e_{8\_3}: \overline{blo_1}$
$e_{9\_3}: t \geq t_{maj} + 1$

Junction functions :
$j_{1\_3}: E_0 = U(1; 2)$
$j_{2\_3}: \ldots$
$j_{3\_3}: \ldots$
$j_{4\_3}: \ldots$
$j_{5\_3}: \ldots$
$j_{6\_3}: \ldots$
$j_{7\_3}: blo_1$
$j_{8\_3}: blo_1$
$j_{9\_3}: e_p = R(T_{zh}; T_{zb}; Z_d; H);\ t_{maj} = t$

Information for fusions of transitions :
…

Equation systems :

F_{1_3}: ...

F_{2_3}: ...

F_{3_3}: ...

$$F_{4\_3}: O\left(\begin{array}{c} T_{zh}^{ip_1}; T_{zh}^{ip_2}; T_{zb}^{ip_1}; T_{zb}^{ip_2}; \Delta P^{ip_1}; \Delta P^{ip_2}; \mathrm{Y}_{o_2;zh}^{ip_1}; \mathrm{Y}_{o_2;zh}^{ip_2}; \\ \mathrm{Y}_{o_2;zb}^{ip_1}; \mathrm{Y}_{o_2;zb}^{ip_2}; \mathrm{Y}_{f;zh}^{ip_1}; \mathrm{Y}_{f;zh}^{ip_2}; \mathrm{Y}_{f;zb}^{ip_1}; \mathrm{Y}_{f;zb}^{ip_2}; Z^{ip_1}; Z^{ip_2}; H; W \end{array}\right)$$

F_{5_3}: ...

F_{6_3}: ...

F_{7_3}: ...

F_{8_3}: ...

where:

$\mathbf{H}; \mathbf{W}$: door dimension, respectively height and width

$\boldsymbol{E_0}$: initial opening state, equals to 1 when closed and 2 when open

$\boldsymbol{e_p}; \boldsymbol{e_{p,s}}$: respectively the energy accumulated in the door and the maximum energy supported

$\boldsymbol{t_{maj}}$: a variable used to fire T_{9_3} every seconds

$\boldsymbol{ip_1}; \boldsymbol{ip_2}$: the numbers of compartments put in communication by the door, respectively 1 (bedroom) and 2 (corridor)

$\boldsymbol{T_{zh}}; \boldsymbol{T_{zb}}$: upper and lower layers temperatures in every compartment composing the system (vector quantity)

$\mathbf{Yo_{2,zh}}; \mathbf{Yo_{2,zb}}; \mathbf{Y_{f,zh}}; \mathbf{Y_{f,zb}}$: dioxygen and fuel mass fractions in upper and lower layers for every compartment composing the system (vector quantity)

$\mathbf{Z}$: thermal interface height in every compartment composing the system (vector quantity)

$\Delta\boldsymbol{P}$: pressure offsets from reference pressure a reference pressure P_0 at ground level for every compartment composing the system (vector quantity)

$\boldsymbol{R}(\dots)$: a CIFI 2009 sub-routine calculating heat transfers in the door (computes the evolution of $\boldsymbol{e_p}$ over time)

$\boldsymbol{O}(\dots)$: a CIFI 2009 sub-routine computing mass and energy fluxes exchanged between the bedroom and the corridor though the doorway

$\boldsymbol{U(x_1; x_2)}$: a function that returns a relative number between x_1 and x_2 with a uniform distribution law

Figure 4 : Example of sub-Petri net representing a door

At the beginning of a scenario, the place P_{1_3} is marked and then the transition T_{1_3} is immediately fired. As a consequence of the firing, the random variable E_0 takes a value for the current scenario (see j_{1_3}). This variable E_0 is used to make vary the door opening initial state. Indeed, if this variable equals 1, then transition T_{2_3} is enabled and fired. A result of its firing, the place P_{3_3} is marked, meaning that the door is initially closed for this scenario. Conversely, if this variable E_0 is equal to 2, then the enabled transition is T_{3_3}. The firing of T_{3_3} brinks a token in place P_{4_3}, indicating that the door is initially opened. Moreover, the two initiate opening states are equally likely (cf. j_{1_1} using function $(\boldsymbol{x_1}; \boldsymbol{x_2})$).

These three transitions T_{1_1}, T_{2_3} et T_{3_3} are therefore used to initialize the network to the current scenario. T_{1_3} is devoted to shooting random variables while T_{2_3} and T_{3_3} are used to initialize the initial state of the object. For further illustration, suppose that the initial state is as follows:

- the door is open (P_{4_3} is marked) ;
- the door is available – it is undamaged (P_{5_3} and P_{6_3} are marked) ;
- the door receives an incoming energy flux from the surrounding gas and heats (P_{8_3} is marked).

Until someone or a door-closer closes the door, place P_{4_3} remains marked. The equation system F_{4_3} is run, which has the effect of calculating the mass and energy fluxes through the doorway. This calculation is performed by calling a CIFI 2009 sub-routine, referred here as O (like Openings). This routine requires input parameters providing information about the characteristics of the gas from both sides of the door (e.g.: gas temperature in both upper layers - see $\mathrm{T}_{\mathrm{zh}}^{\mathrm{ip}_1}$ and $\mathrm{T}_{\mathrm{zh}}^{\mathrm{ip}_2}$) and characteristics of the door itself (e.g.: its dimensions - see H and W). One may notice that variables whose time-evolution is predicted by the CIFI 2009 gas model appears as image variables in the sub-net (see $T_{zh}; T_{zb}; \Delta P; Yo_{2,zh}; Yo_{2,zb}; Y_{f,zh}; Y_{f,zb}; Z$).

Assume that a door closer operates: it is the firing of T_{5_3}. This closure may only take place if the door is still undamaged, that is to say if the Boolean variable blo_1 is equal to false (see e_{5_3}). The door, if it remains undamaged, may then be opened during the scenario (e.g. by an evacuee), which corresponds to the firing of T_{4_3}. In this case, the door closer closes the door again (shooting T_{5_3}).

In parallel, the transition T_{9_3} is fired every second (see e_{9_3}). This transition serves to calculate the energy absorbed in the door. This calculation is performed in junction function j_{9_3}, more specifically by the call of a CIFI 2009 sub-routine named R (as Ruin) in Figure 4. Therefore, value of the variable e_p increases over time. If this variable exceeds $e_{p,s}$ (the threshold of ruin), then the door is destroyed: the transition T_{7_3} is fired. As a consequence, the Boolean variable blo_1 becomes true (see j_{7_3}). Transitions T_{4_3} and T_{5_3} are thus inhibited (see e_{4_3} and e_{5_3} and which require variable blo_1 must be false to enable transitions T_{4_3} and T_{5_3}). This means that any other object can handle a damaged door.

Furthermore, this door may also be destroyed by another system object (e.g.: by firefighters). This case corresponds to the firing T_{8_3}. In the same way, if T_{8_3} is fired, the Boolean variable blo_1 becomes true (see j_{8_3}) and transitions T_{4_3} and T_{5_3} are inhibited.

Finally, considering that the door destruction leaves the doorway wide open, it is necessary to bring the token in place P_{4_3} in order to calculate the appropriate mass and energy fluxes through the doorway. This role is fulfilled by transition T_{6_3}. Indeed, this transition is enabled when the closed door undergoes ruin; that is to say that P_{3_3} is marked and that the Boolean variable blo_1 becomes true (see e_{6_3}).

6 Conclusion

A new fire safety engineering tool called SCHEMA-SI (Stochastic Computation and Hybrid Event Modelling Approach – Sécurité Incendie) was built. This tool relies on the hybridisation of a two-zone model and Petri nets. This paper summarize both sub-models and describes the hybridising method of the tool is described in this paper. As this tool permits to generate thousands of fire scenarios, it was already used for fire risk assessment [20], to evaluate the performance of fire safety strategies [21] and for fire reconstitution [22]. SCHEMA-SI may help fire protection engineers and building officials to select acceptable solutions for designing a fire safety system. It may be also used to optimise the most cost effective design solution. In the future, further applications of the SCHEMA SI tool are required. The validation of SCHEMA-SI for these applications may help the introduction and use of performance/objective based codes in France.

References

[1] Friedman, R. (1992). Survey of computer models for fire and smoke. Journal of Fire Protection Engineering, 4 (3), 81-92.

[2] Peacock, R., Jones, W., Forney, G., & Reneke, P. (2009). CFAST – Consolidated Model of Fire Growth and Smoke Transport. NIST.

[3] Franssen, J.-M. (2004). SAFIR: A Thermal/Structural Program Modeling Structures under Fire. Engineering Journal, A.I.S.C., 42, 143-158.

[4] Gwynne, S., Galea, E. R., Lawrence, P. J., & Filippidis, L. (2001). Modelling occupant

interaction with fire conditions using the buildingEXODUS evacuation model. Fire Safety Journal, 36(4), 327-357.

[5] Olenick, S., & Carpenter, D. (2003). An Updated International Survey of Computer Models for Fire and Smoke. Journal of Fire Protection Engineering, Vol. 13, 87-110.

[6] Jinhua, S., & Guanquan, C. (2008). Quantitative Assessment of Building Fire Risk to Life Safety. Risk Analysis, 28, 615–625.

[7] He, Y. P., Horasan, M., Taylor, P., & Ramsay, C. (2002). Stochastic modelling for risk assessment., (pp. 333–344).

[8] Hasofer, A. (2009). Modern sensitivity analysis of the CESARE-Risk computer fire model. Fire Safety Journal, 44(3), 330-338.

[9] Yung, D., Hadjisophocleous, G., & Proulx, G. (1999). A Description of the probabilistic and deterministic modelling used in FIRECAM. International Journal on Engineering Performance-Based Fire Codes, 1, 18-26.

[10] Bénichou, N., Kashef, A., Reid, I., Hadjisophocleous, G., Torvi, D., & Morinville, G. (2005). FIERAsystem: a fire risk assessment tool to evaluate fire safety in industrial buildings and large spaces. Journal of Fire Protection Engineering, 15(n°3), 145-172.

[11] Peacock, R., Jones, W., Forney, G., Portier, R., Reneke, P., Bukowski, P., et al. (1994). An Update Guide for HAZARD I Version 1.2. NIST.

[12] Hostikka, S., & Keski-Rahkonen, O. (2003b). Probabilistic simulation of fire scenarios, p. 301–311. Nuclear Engineering and Design, 224.

[13] Robert, C., & Casella, G. (2004). Monte Carlo Statistical Methods. Springer.

[14] Bodart, X., Curtat, M., Blay, D., Tuhault, J.-L., & Botlan, Y. L. (1997). Study of fan-powered extraction of smoke in a room where a fire develops: full-scale experiments and computer simulations using a zone model and a CFD model. Revue Générale de Thermique, 36 (5), 372-397.

[15] Jensen, K. (1981). Coloured Petri Nets and the Invariant Method. Computer Science, 14.

[16] Genrich, H. J., & Lautenbach, K. (1981). System Modelling with High-Level Petri Nets. Theoretical Computer Science, 13.

[17] Villani, E., Valette, R., & Miyagi, P. (2007). Modelling and Analysis of Hybrid Supervisory Systems : A Petri Net Approach. Springer.

[18] Petri, C.-A. (1962). Kommunikation mit Automaten. Institut für Instrumentelle Mathematik.

[19] Zaytoon, J. (2001). Systèmes Dynamiques Hybrides. Hermes Science.

[20] Muller, A., Demouge, F., Brilhac, J.-F., Fromy, P., & Jequirim, M. (2008). Sécurité Incendie : Développement d'une Approche de Modélisation Stochastique Hybride. *Congrès Maîtrise des Risques et Sureté de Fonctionnement, dit Lambda-Mu.* Avignon.

[21] Muller, A. (2010). *Développement d'une méthode de modélisation pour l'évaluation de la performance de stratégies de sécurité incendie.* Université de Haute-Alsace. Thèse de l'université de Haute-Alsace.

[22] Muller, A., Demouge, F., Fromy, P., & Vantelon, J.-P. (2010). The use of a stochastic fire safety engineering tool for fire scene reconstruction. *Interflam Congress.* Nottingham.