

# Dynamic flowgraph methodology and its applications

Authors: Tero Tyrväinen

Confidentiality: Public

<b>Report's title</b>	
Dynamic flowgraph methodology and its applications	
<b>Customer, contact person, address</b>	<b>Order reference</b>
VYR	SAFIR 4/2016
<b>Project name</b>	<b>Project number/Short name</b>
Probabilistic risk assessment method development and applications	102420/PRAMEA
<b>Author(s)</b>	<b>Pages</b>
Tero Tyrväinen	18/
<b>Keywords</b>	<b>Report identification code</b>
Dynamic flowgraph methodology, reliability, digital systems	VTT-R-03364-16
<b>Summary</b>	
<p>Dynamic flowgraph methodology (DFM) is method for the reliability analysis of dynamic systems with time-dependencies and feedback loops. As in fault tree analysis, the aim of DFM is to identify which conditions can cause a top event, which can be, for example, the system's failure. DFM has been most often applied to different digital control systems. One reason for this is that a DFM model can represent the interactions between a control system and the controlled process. Components of DFM models are analysed at discrete time points and they can have multiple states. The reason for the development of DFM is that traditional methods, such as fault tree analysis, can describe the system's dynamic behaviour only in a limited manner. DFM can more accurately represent system's evolution in time.</p> <p>This report gives an overview of the DFM method and presents the applications of DFM that are found in literature. The application areas include digital control and safety systems in nuclear power plants, space systems, hydrogen production plants, human performance, networked control systems and field programmable gate arrays. In most of the applications, DFM has been used to analyse how control system failures can cause some physical variable, e.g. water level or pressure, to have too low or high value. Generally, DFM has been found useful within the application areas. Most of the presented models have been quite moderately sized, though larger models exist too.</p>	
<b>Confidentiality</b>	Public
Espoo 24.1.2017	
<b>Written by</b>	<b>Reviewed by</b>
	
Tero Tyrväinen Research Scientist	Ilkka Karanta Senior Scientist
	<b>Accepted by</b>
	
	Eila Lehmus Research Team Leader
<b>VTT's contact address</b>	
VTT, PL 1000, 02044 VTT	
<b>Distribution (customer and VTT)</b>	
SAFIR reference group 2	
<p><i>The use of the name of VTT Technical Research Centre of Finland Ltd in advertising or publishing of a part of this report is only permissible with written authorisation from VTT Technical Research Centre of Finland Ltd.</i></p>	

## Contents

---

1. Introduction.....	3
2. Dynamic flowgraph methodology .....	3
3. Applications .....	6
3.1 Digital systems in nuclear power plants .....	6
3.1.1 Digital feedwater control system .....	6
3.1.2 Control system of a pressurizer.....	8
3.1.3 Programmable electronic safety system.....	9
3.1.4 Borated feedwater control system .....	9
3.1.5 Steam generator level control system .....	10
3.1.6 Emergency core cooling system .....	10
3.1.7 Simple feedwater system .....	10
3.2 Space systems .....	11
3.2.1 Digital flight control system .....	11
3.2.2 Ion-propulsion system.....	11
3.2.3 Thermal control system.....	12
3.3 Networked control systems.....	12
3.4 Hydrogen production plants .....	13
3.4.1 Copper-chloride cycle .....	13
3.4.2 Super critical water reactor integration .....	13
3.5 Human performance.....	13
3.6 Field programmable gate arrays .....	14
3.7 Water supply process control.....	15
4. Conclusions .....	15
References.....	16

## 1. Introduction

---

Dynamic flowgraph methodology (DFM) is a method for the reliability analysis of dynamic systems with time-dependencies and feedback loops. It was developed by C.J. Garrett, S.B. Guarro and G.E. Apostolakis in 1990's. The first journal article and the main reference presenting DFM is "The dynamic flowgraph methodology for assessing the dependability of embedded software systems" published in 1995 [1]. As in fault tree analysis, the aim of DFM is to identify which conditions can cause a top event, which can be, for example, the system's failure. A DFM model is a graph representation of the analysed system. The nodes of the graph represent system's variables, e.g. physical and software variables, and the edges between them represent causal and other relationships between the variables. Components of DFM models are analysed at discrete time points and they can have multiple states. The reason for the development of DFM is that traditional methods, such as fault tree analysis, can describe the system's dynamic behaviour only in a limited manner. DFM can more accurately represent system's evolution in time.

DFM has been most often applied to different digital control systems that include both hardware and software components. One reason for this is that a DFM model can represent the interactions between a control system and the controlled process. DFM supports the modelling of multi-state components, which is an advantage in modelling digitally controlled systems because their components generally do not behave in binary manners. Another advantage of DFM is that only one model is needed to represent the complete behaviour of a system and therefore different states of the system can be analysed using the same model [2].

The result of DFM analysis is a set of prime implicants [3, 4]. A prime implicant is a minimal combination of basic events and other conditions that is sufficient to cause the top event. DFM analysis considers at which time points events have to occur to cause the top event. Compared to static fault tree analysis, DFM provides more accurate information about the development of accident scenarios and makes more accurate probability calculations possible.

Section 2 describes the DFM methodology. Section 3 presents the applications of DFM that are found in literature. The application areas include digital control and safety systems in nuclear power plants, space systems, hydrogen production plants, human performance, networked control systems and field programmable gate arrays. Section 4 concludes the report.

## 2. Dynamic flowgraph methodology

---

A DFM model is a directed graph that consists of nodes representing the system's components and variables, and edges representing causal and other dependencies between nodes. A node can have a finite number of states and the state of a node is determined either by a probability model or by states of its input nodes at specified time steps relative to the time step considered. Input dependencies of a node are represented in a decision table which is an extension of a truth table. Decision tables can be constructed based on empirical knowledge on the system, physical equations, simulations, expert judgement, software design or software code.

Figure 1 shows an example of a DFM model based on a tank system with a digitally controlled valve, and Table 1 gives an example of a decision table. A tank gets water from an infinite water source. The outflow from the tank is regulated by a valve, which is controlled by a digital controller based on measurement of the water level. In the model, node C represents the functional state of a valve, N represents water level measurement value ( $N=-1$

means that the water level is below the reference value,  $N=1$  that it is above the reference value, and  $N=0$  that it is within tolerance limits of the reference value) and  $T$  represents water level. Nodes  $F$  and  $R$  determine whether the valve and the water level measurement are failed and they change states by a probability model. Each row of the decision table represents a state combination of input nodes ( $F$ ,  $N$  and  $C$ ) and the output column determines to which state of the output node  $C$  each state combination of input nodes leads to. The time lag row determines the delays in the dependencies between the input nodes and the output node. The time lags are also seen in Figure 1. In Table 1, node  $C$  depends on its own state at the previous time step because the time lag is 1.

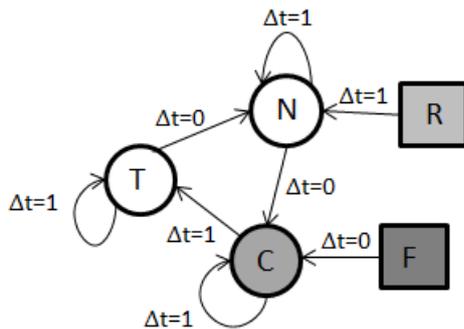


Figure 1: A DFM model with five nodes.

Table 1: The decision table of component  $C$ .

Node	Output	Inputs		
	C	F	N	C
Time lag		0	0	1
0	0	0	-1	0
0	0	0	-1	1
0	0	0	0	0
1	0	0	0	1
1	0	0	1	0
1	0	0	1	1
0	1	1	-1	0
1	1	1	-1	1
0	1	1	0	0
1	1	1	0	1
0	1	1	1	0
1	1	1	1	1

The primary target of DFM is to identify prime implicants of the top event. A prime implicant is a minimal combination of conditions that is sufficient to cause the top event. In DFM, these conditions are represented by literals. In this context, a literal is triplet consisting of a variable  $V$ , state  $s$  and time point  $-t$ , and denoted as  $V_s(-t)$ . Hence, prime implicants of DFM can be understood as multi-state and timed minimal cut sets. The mathematical definition of prime implicants is presented and discussed in [4].

The top event is also defined as a set of literals. The analyst can freely choose any top event. Therefore, it is possible to analyse several top events in parallel in the same analysis, and both success and failure scenarios can be analysed.

In DFM, there are two types of nodes: deterministic nodes and stochastic nodes. The state of a deterministic node is determined by its input nodes through a decision table. The state of a stochastic node is determined by a probability model. At the initial time step, a deterministic node behaves like a stochastic node. Implicants of a top event can contain initial states of deterministic nodes and states of stochastic nodes at any time step.

A DFM model is typically analysed by tracing event sequences backwards from effects to causes. Deductive analysis starts from the top event. The model is traced backwards in the cause-and-effect flow to identify what initial states of deterministic nodes and states of stochastic nodes produce the top event. The process ends when the initial time step is reached.

DFM models can also be analysed inductively (forward in time) by simulating the model with particular initial conditions. All the possible consequences of the system's initial or boundary conditions are generated. The initial or boundary conditions can either be desired or undesired states. If these conditions are desired states, an inductive analysis can be used to verify system requirements, meaning that normal operation under normal conditions does not lead to undesired states. If these conditions are undesired states, inductive analysis can be used to verify the system's safety behaviour. Inductive analysis can, for example, be used to analyse prime implicants identified in deductive analysis in greater detail, and examine the effects of mitigation actions.

Many DFM papers, e.g. [1], use concepts such as process node, condition node, causality edge, condition edge, transfer box and transition box. The difference between process nodes and condition nodes only comes from the modelling philosophy. From technical point of view, there is no difference. Transfer boxes correspond to decision tables without time lags, and transition boxes correspond to decision tables with time lags. Causality edges connect process nodes, and condition edges connect condition nodes to process nodes via transfer or transition boxes.

There are two known DFM software tools, Dymonda [5] and Yadrat [6]. Dymonda has been developed by the original developers of DFM. It solves the graph model by transferring it to a timed fault tree, which is a combination of static fault trees representing different time steps, and from which the prime implicants can be generated using regular fault tree algorithms. Yadrat has been developed by VTT. It transforms the DFM model into a binary decision diagram from which the prime implicants are solved. Different tools use slightly different specifications and terminology. Dymonda follows the "official" DFM specifications [7]. Yadrat can be considered as an alternative interpretation of the methodology. Despite the differences, the same deductive analyses can be performed using both tools. Yadrat does not support inductive analysis.

In the computation of the top event probability in DFM, the basic idea is similar to the computation of the top event probability in fault tree analysis. In DFM, the top event probability is calculated based on the prime implicants and the probabilities of the literals. The determination of the probabilities for different states and time steps can be challenging, especially if there are functional dependencies between literals. The probabilistic

computations have been addressed very little in literature, and there is no paper specifically dedicated for the subject. Some probability models have been referred, such as exponential model used in [8]. For top event probability computation, ordinary upper bound algorithms used in fault tree analysis can be applied in DFM too. More accurate top event probability algorithms have also been developed, such as the algorithm presented in [9] and the algorithm that is mentioned in [7].

Risk importance measures for DFM have been studied in few references [8, 10, 11]. Reference [8] presents dynamic versions of Fussel-Vesely and risk increase factor. These dynamic risk importance measures are formulated for the states of components and they take the time-aspect of DFM into account.

Common cause failure modelling has also been studied in the DFM context [12]. In the models developed in [12], the components can fail at different time points due to a common cause, i.e. the failures do not need to be simultaneous.

### 3. Applications

---

This section presents various DFM models that are found in literature. The analysed systems and the models are briefly described. The results are also discussed if they are presented comprehensively enough in the references. Many references do however not give the details of the models and address the results very little. Anyhow, reader needing more detailed information should read the references. The purpose of this section is just to give some ideas of what kind of analyses have been performed using the DFM.

#### 3.1 Digital systems in nuclear power plants

The reliability analysis of digital systems is considered one of the biggest challenges in modern nuclear power plant PRA. Application of traditional static methods, such as fault trees, to digital systems is very restricting. Fault trees cannot capture the dynamic interactions of such systems well. NUREG/CR-6901 [13] has identified DFM as one of the promising methods for the reliability analysis of digital I&C systems. DFM has been considered effective in modelling dynamic interactions, such as delays, memories, logic loops and system states [14]. Interactions can, for example, lead to coupling of events, such as opening of valve and starting of pump, and therefore, have a significant effect on the system's reliability. Multi-state logic is also a benefit because the behaviour of software controlled systems is usually non-binary.

##### 3.1.1 Digital feedwater control system

In [15, 16], DFM was applied to a digital feedwater control system (DFWCS) used in pressurised water reactors (PWR). The aim of the study was to compare DFM and the Markov/cell-to-cell mapping technique.

The analysed system is presented in detail in [15]. The purpose of the DFWCS is to keep the water level in steam generators within set limits. The model contains two steam generators, both of which have their own digital controller. A controller regulates the water flow to a steam generator by controlling a feedwater pump, a main feedwater regulating valve and a bypass feedwater regulating valve.

A controller has a main computer and a back-up computer. Computers were modelled with three macro states:

- Both computers operating normally
- One computer down but can be recovered

- One computer down and cannot be recovered

Macro states have also internal states: operating, loss of one input, loss of both inputs, computer down and arbitrary output. If a computer is down in the macro state, these internal states represent the state of the second computer. If both computers fail, the controller continues sending the last valid value to the actuator if the failures are detected. The controller failure modes that are included in the model are:

- Arbitrary output
- Output high
- Output low
- Loss of communications

Mechanical failures of valves and pumps are modelled so that they are stuck in their current state when a failure occurs.

The scenario that was analysed was a change in power operation from 70% (of the full power) to 78% and back to 70%. This transient induces a challenge for the DFWCS to keep the water level in the steam generator within set limits.

The data used in the probabilistic calculations was based on fault injection experiments, operating experience and generic data bases. The modelling of physical phenomena was based on deterministic simulations.

The following DFM analysis is presented according to [16]. The model in reference [15] has partially different nodes, structure and modelling philosophy. The model of [15] was likely developed further after the report was published, and the model of [16] is the result of that development.

The components that are included in the DFM model are the main computer, back-up computer, main flow valve, bypass flow valve, feed pump and PID controller. Valve nodes represent both the physical component and the controller. Components are mostly modelled using three nodes representing the state of the component, the previous state of the component and the state transition. The state transition nodes are stochastic. This modelling style was probably chosen because DFM was compared to the Markov/cell-to-cell mapping technique, where similar state transitions need to be defined. Some components are also associated with some other nodes, such as a node representing pump speed.

The model contains nodes for steam generator water level, feed flow, steam flow and reactor power. Process variables are typically discretised to around five states. Flow demands are calculated based on the process variables. Power to the controllers is modelled with one node as well as the power to the computers. Steam generator water level measurement error is also included in the model. All failures were modelled as non-repairable, i.e. the component remains in the failure state for all the remaining time steps once it has failed. In total, the model contains 29 deterministic nodes and 10 stochastic nodes.

The top events that were analysed were too high water level in the steam generator and too low water level in the steam generator. There is a risk for the water level being too low when the reactor power is increased, and there is a risk for the water level being too high when the reactor power is decreased. The model was traced backwards only one time step for both top events. For water level too low, time step 0 is the time when the reactor power is 78% and time step -1 is the time when the reactor power is 70%. The duration between the time steps is eight hours. Respectively, for water level too high, time step 0 is the time when the reactor power is 70% again and time step -1 is the time when the reactor power is 78%.

For too low water level, 1197 prime implicants were generated. According to the prime implicants, the biggest failure contributor is that the main feedwater valve is stuck in position 70-74%. The failure of the controller power and the failure of the computer power appeared also in some of the most important prime implicants. The top event probability was  $4.19E-4$ .

For too high water level, 138 prime implicants were generated. The dominant failure contributor is that the main feedwater valve is stuck in position 78%. The top event probability was  $3.34E-4$ .

The results of DFM and the Markov/cell-to-cell mapping technique were consistent. An approach utilising both DFM and Markov analysis is proposed. It is suggested that DFM could first be used to identify prime implicants. Then, inductive Markov analysis could be performed to validate the prime implicants and to examine their sensitivity to variations of initial conditions.

The study suggests that DFM can be used as a supplementary approach in probabilistic risk assessment (PRA) to model digital I&C systems. In [15], it is presented how a DFM model can be integrated into a PRA model. The integration is quite simple if the system modelled using DFM has no dependence with other systems modelled in PRA. In that case, the top event probability of DFM can directly be used in PRA. However, taking into account dependencies between a DFM model and a fault tree model is not simple. The reference presents how multi-state logic is transformed into binary logic and time step information is ignored. The integration of a DFM model to PRA was later studied in VTT's research report [17], where the integration was especially researched from PRA software point of view. Several different cases of integration were investigated, and prime implicants of DFM were successfully transformed into a fault tree form. Events that are common to DFM and fault trees require some special handling in naming and possibly also in modelling.

### 3.1.2 Control system of a pressurizer

A simplified digital control system of a PWR pressuriser was modelled using DFM in [14]. The purpose of the system is to maintain the pressure in the pressure vessel in the target value. If the pressure deviates from the target value, the pressure can be increased using heaters or decreased using sprays, a relief valve or a safety valve.

In the model, heaters and sprays can be failed either on or off. Pressure sensors can be failed to a high or low value. Valves can be failed opened or closed. The pressure is discretised into seven states.

The analysed top event was the pressure being at very low level. It is stated that the top event probability is very low, but no comprehensive analysis of results is given.

It is briefly presented how the DFM results can be incorporated into PRA. Prime implicants are converted into a fault tree, and state and time step information is included in basic event names.

Pinto et al. have applied a methodology that comprises DFM and a Technique for human error analysis (ATHEANA) to the same digital control system of a PWR pressuriser [18]. The model covers the control system and its interactions with the process and operator. Operator errors and factors affecting human performance were included in the model.

Two top events were analysed for this model: very high pressure and very low pressure. The model was traced backwards only one time step. For very high pressure, 374 prime implicants were generated, and for very low pressure, 90 prime implicants were generated. Some prime implicants with human errors and performance shaping factors were highlighted and analysed further with ATHEANA.

### 3.1.3 Programmable electronic safety system

Houtermans et al. [19] demonstrate how DFM can be utilised in the design and verification phase of digital safety-related systems. The example system controls the temperature of a tank by regulating a valve. The system takes three signals from sensors as inputs and outputs one actuation signal. The software and hardware are modelled in a fairly detailed manner. The model contains DFM representations of input and output channels, input and output circuitry, bus communications and controller including application software and RAM. The controller's interactions with the operator and a basic process control system (BPCS) are also modelled. The model includes 28 deterministic nodes and 18 stochastic nodes. The level of detail reflects identifiable function blocks that fail as an entity.

The analysed top event was that the valve is not opened on demand. The model was traced backwards 14 time steps. 1190 prime implicants were generated. 83 prime implicants included only one literal. A prime implicant, where an output channel is stuck in high value and causes the top event, is highlighted. Another prime implicant indicates that a failure of the BPCS alone can cause the top event, which is so because the safety system depends on the BPCS. It is recommended that the safety system should be made independent of the BPCS. Also, one prime implicant indicates that the top event occurs if an operator sets the temperature limit too high. Therefore, some safety procedures should ensure that the operator does not make that mistake.

Another analysis was performed so that the applications software was modelled with one node only (originally modelled with 13 nodes) while the model was otherwise the same. A programming error was injected to this software node ('>' used instead of '<'). The model was analysed inductively so that all hardware was operating normally. The result was that the valve was not opened on demand because the software did not perform as intended (due to the error).

Verification of design requirements, failure analysis and defining software test cases were identified as applications of DFM in this context.

### 3.1.4 Borated feedwater control system

In [1], Garrett, Guarro and Apostolakis present DFM analysis of a simple control system that controls the flow of borated water to a nuclear reactor. The system controls two valves, one assigned to a water tank and one assigned to a boric acid tank. The valves are controlled based on measurements coming from sensors that measure the flowrates. The required flowrate to the nuclear reactor and boron concentration depend on the reactor power level. Therefore, the controller must calculate the flowrates needed from tanks based on the power level.

The DFM model is simplified. It is assumed that the sensors can only fail either to high value or to low value. The valves have five states representing different degrees of openness, but they fail only to fully open state or to fully closed state. The dependence of the required flowrates from the reactor power level is modelled including time delays needed for computation (50 seconds in total). The required position of a valve is determined based on the required flowrate, the measured flowrate from the tank and the measured previous position of the valve. The time delay in the computation of the positions of the valves is also 50 seconds. The model contains 16 deterministic nodes and 4 stochastic nodes.

The analysed top event was the flowrate from the boric acid tank being too high causing the reactor shut down. The prime implicants are however not presented and the results are not significantly analysed. The purpose of the example was to demonstrate DFM modelling.

### 3.1.5 Steam generator level control system

Yau, Apostolakis and Guarro present DFM analysis of a PWR steam generator level control system in [3]. The system contains a digital controller which regulates a valve based on data from a feed flow sensor, a level sensor and a steam flow sensor. The details of the DFM model are not given; the paper focuses more on the results.

The analysed scenario was a steam generator overflow in the context where a fault was injected into the control system specification. The model was traced backwards only one time step. 10 prime implicants were generated and the injected fault was identified based on two prime implicants that did not contain any component failures contributing to the top event. Four prime implicants contained the level sensor showing low readings while the real level was high. Another four prime implicants included the valve being stuck fully open. The results also demonstrated the non-coherent reliability analysis of DFM. For example, in all prime implicants, the pump needed to be working so that the top event could occur, because without the pump, the water level could not increase to the overflow state.

### 3.1.6 Emergency core cooling system

An emergency core cooling system model was used demonstrate DFM analysis in [6, 8]. The purpose of this system is to provide adequate water cooling of a reactor core if the ordinary cooling system is not functioning. An on-off system controls the water level in the pressure vessel by controlling pumps and regulation valves. Sensors measure the water level and the pressure. The water level can decrease due to evaporation. If the water level is measured to be too low, the pumps are started, the valves are opened, and more water is pumped into the pressure vessel until an upper limit is reached. The valves can be opened only if the pressure is measured to be low.

The DFM model contains one pump line that includes four components: a water level sensor, a pressure sensor, a regulation valve and a pump. Each component is modelled using a deterministic node which represents the functional state of the component and a stochastic node which determines whether the component is failed. Pump leakage signal is also included in the model. The model includes deterministic nodes to represent the water inflow and the reactor water level as well as the signals between the sensors, the control logic and the actuators.

The analysed case was that the water level is low four time steps in a row. The model was traced backwards five time steps because earlier experiences had shown that all the relevant prime implicants can be identified using this time frame and same patterns are only repeated in prime implicants using a longer time frame. 338 prime implicants were generated in [8]. In [6], the number was different likely because the model was slightly different. In [8], risk importance measure values were calculated for different failure events of the model. Pump leakage signal, pump failure and valve failure to closed state were identified as the most important events. In addition, many prime implicants included a condition that a pressure measurement needed to work at particular time step so that the top event could occur.

### 3.1.7 Simple feedwater system

A simplified DFM model of the feedwater system of Olkiluoto 1&2 boiling water reactor nuclear power plant units was presented in [20]. The model analyses restart after a short term loss of offsite power transient. The model contains 23 nodes representing flow rates, water level in the pressure vessel, water level measurement, control logic, positions of valves and pump speed. 2082 prime implicants were generated for too low water level in the pressure vessel, and some non-trivial failure modes of the system were identified in the analysis.

## 3.2 Space systems

Space systems utilise also often digital control systems. Hence, their reliability analysis has partly similar challenges as the reliability analysis of digital systems in nuclear power plants. Because of this, DFM has been applied to space systems since 1990's.

### 3.2.1 Digital flight control system

Yau, Guarro and Apostolakis [21] demonstrated DFM analysis using the digital flight control system of Titan II Space Launch Vehicle which is a space rocket. The digital flight control system stabilises the vehicle and controls its altitude. The system also establishes the flight path by implementing commands from a guidance system. The digital flight control system contains:

- Missile guidance computer and flight control software
- Altitude rate sensing system
- Inertial measurement unit
- Hydraulic actuators

The flight control software contains a major cycle and a minor cycle. The major cycle is used to control the general flight directions. It lasts for 1 second. The minor cycle is used to more urgent calculations and lasts for 40 ms. The software also reads inputs from sensors and sends outputs to the actuators every 5 ms.

The DFM model contains both hardware and software part. The model is quite complex, and the details are difficult to understand. The hardware part contains around twenty nodes and the software part contains over fifty nodes. The results of the analysis are not presented except related to alternative DFM computation techniques.

### 3.2.2 Ion-propulsion system

In [22], DFM was applied to a spacecraft ion-propulsion system, which is used to reach the orbit of a distant planet. The main components of the system are five thrusters and a tank. The system is used in phased-missions, and in different missions, different number of thrusters is needed.

Two DFM models were built in the study: a high level model comprising three missions and a low level model for detailed modelling of the thrusters. The high level model contains a block for each mission, thruster and propellant distribution line. Each block consists of only two nodes. Missions can be affected by failures of thrusters and leakage in a propellant distribution line. The model specifies the number of required thrusters for each mission.

The high level model was traced backwards 13 time steps containing three steps for the first mission, two for the second and seven for the third. The prime implicants contained combinations of failures of the propellant distribution lines and thrusters at different time steps. It is highlighted that the dependencies between different missions can be modelled using DFM because the failure histories of components can be taken into account in the analysis.

The low level model includes three failure modes for each thruster: failure to start, failure to operate and failure to shut down. These failure modes can be caused by failures of propulsion power unit, two ion engines or two propellant valves. Common cause failures between ion engines are modelled using separate nodes.

The low level model was traced backwards three time steps in a simplified configuration. The prime implicants contained combinations of failures of ion engines and common cause failures between ion engines at different time steps.

The plan of the authors was to continue the validation of the models. The final models would be analysed so that the prime implicants of the high level model would be identified and then analysed using the low level model. The analysis was divided into two levels to avoid combinatorial explosion.

### 3.2.3 Thermal control system

A thermal control system (TCS) was analysed using DFM in [23]. The system maintains the temperature of a rack (experiment platform) within the accepted range by removing waste heat which is generated by scientific experiments. It contains components such as valves, pipes, cold plates, pumps and controllers. Dynamic interactions between components, e.g. hardware and software, cannot be modelled using static methods, and therefore, DFM was chosen for the reliability analysis.

The DFM model includes 19 deterministic nodes and 15 stochastic nodes. Deterministic nodes represent temperatures in different places, water flows, software parameters and valve actions. Stochastic nodes represent thermal loads of water cooling and air cooling, valves, software, hardware, actuator controlling systems and remote operations.

The top event of the analysis was high level of the temperature of the rack outlet water. The model was traced backwards only one time step. 21 prime implicants were generated. The prime implicants are not discussed comprehensively, but it is stated that some prime implicants are not realistic and that the time lags between valve actions are difficult to determine. Anyhow, valves failing closed are highlighted as causes for the top event.

## 3.3 Networked control systems

Al-Dabbagh and Lu [2, 24] have studied DFM modelling of networked control systems (NCS). In NCSs, sensors, controllers and actuators are connected via communication networks. DFM is considered suitable to NCS modelling because it is able to capture the behaviour and interaction of the hardware, the software and the communication network in the NCSs, and enables time-dependent and multi-state modelling.

For a networked control system, modelling of degradation in performance is interesting, and most often time delays are reasons for the degradation. If a message, e.g. between controller and actuator, is not transmitted in time, the actuator uses the information of the message from the previous cycle.

First, a generic DFM model was built to analyse the time delays in the networked control system. The idea is that this model is a ready-made DFM block that can be incorporated into wider DFM models. The model includes seven deterministic nodes and eight stochastic nodes. The nodes mostly represent different delays, such as pre-processing time delay, post-processing time delay, transmission time and waiting time. The total delay is calculated from different delays. Also, network availability and the states of both source and destination hardware and software are included and their effects on the delays are modelled.

The model was analysed with the unavailability of communications as the top event. The model was traced backwards only one time step. Five prime implicants were generated. Waiting time was identified to be an important factor for the system's performance, and its reduction would therefore be beneficial. In addition, the availability of the communication links and processors was identified as important factor.

The generic DFM block was utilised in modelling a simple main stream flow system, where the flow is controlled by regulating a valve based on data from a sensor. The generic block was used to model the communication between the sensor and the controller, and the communication between the controller and the valve. The model contains ten deterministic nodes and seven stochastic nodes in addition to those that are in the generic block. Results of this model were not presented.

## 3.4 Hydrogen production plants

### 3.4.1 Copper-chloride cycle

Al-Dabbagh and Lu [24, 25] used DFM in modelling of a copper-chloride cycle of a nuclear-based hydrogen production plant. In the copper-chloride cycle, water is decomposed into hydrogen and oxygen. The cycle consists of five interconnected reactor units: hydrogen reactor, electrochemical cell, spray drying unit, fluidized bed and oxygen reactor. The thermochemical process of copper-chloride cycle is controlled by a networked control system which was developed in the study. The networked control system contains a plant display system and five control partitions, each for controlling and monitoring one unit.

The DFM model is one of the largest found in the literature. Each connection between two units is modelled using two nodes: transmission node and recipient node. A transmission node can e.g. represent a production requirement that is sent from a unit to another, and the corresponding recipient node represents then the received production requirement. All units as well as the plant display system and the communication network have their own DFM blocks. The DFM block of the communication network is the same that was presented in the previous section. It affects each recipient node so that the recipient node uses the previous state of the transmission node if there is a delay in communication.

The DFM blocks of individual units contain nodes for modelling components such as valves, sensors, pumps, motors and heat exchangers, and process variables such as gas and liquid flows, temperatures and amounts of produced materials. In total, the model includes at least one hundred nodes. The results of the model are not presented.

### 3.4.2 Super critical water reactor integration

F. Ahmed [26] has applied DFM to integration of copper-chloride cycle hydrogen and oxygen units with super critical water reactor primary heat transport cycle. The DFM model contains nodes to represent components such as sensors, controllers, pumps and valves, and process parameters such as flows, levels and power. Considered major failure modes include hydrogen unit failure, oxygen reactor failure, high pressure/temperature failures, super-heated steam loop failure and improper steam/condensate supply to different components. The details of the model and results are not presented.

## 3.5 Human performance

NUREG/CR-6710 [27] applied DFM to human error analysis of nuclear power plant operator teams. The report states that, in this context, the DFM model should contain the plant, the instruments that provide information, the cognitive behaviour of team members and the interactions between team members. Three stages of DFM analysis are suggested:

1. Verification of the model by simulations
2. Identification of the prime implicants of a human failure event
3. Study of the effects of the human errors by simulating the model with prime implicants as boundary conditions

DFM blocks that model different human performance related effects are presented, including

- Misleading instrumentation
- Errors in monitoring and detection
- Operator assessments
- Operator's mental model of plant behaviour
- Operator diagnosis state
- Individual operators and their communication
- Response planning activities

A DFM model was constructed for a shutdown cooling scenario. The main process variable analysed is reactor coolant system (RCS) level, which depends on high pressure safety injection system and containment spray. The operator must decide which system to use based on information on RCS level and its changes. Several different human failure and communication failure scenarios are included in the model. The model contains 18 deterministic nodes and 10 stochastic nodes.

The analysed top event was an inadequate level of RCS. The model was traced backwards three time steps. The length of a time step was assumed to be one minute. 78 prime implicants were generated. Prime implicants typically include at least one failure at both time steps -3 and -2. It is concluded that if operators make an error, they still have time to notice and correct it. The top event occurs only they make another error.

Another DFM model was built in the same study for an interfacing system loss of coolant accident. In this scenario, a leakage from the reactor coolant system to a residual heat removal (RHR) occurs. The model contains nodes for the leakage, RHR pressure, RHR piping integrity, indications on the control board, high pressure alarm and operators' performance. In total, the model includes 12 deterministic nodes and 5 stochastic nodes.

The analysed top event was that the operators fail to isolate the leak before the RHR piping bursts. The model was traced backwards three time steps. 25 prime implicants were generated. Typically these prime implicants are combinations of events where the operators fail to detect the leak early and fail to isolate the leak after they finally detect it.

### 3.6 Field programmable gate arrays

McNelles, Zeng and Renganathan [28] modelled a field programmable gate arrays (FPGA) based test system using DFM. FPGAs are an alternative for creating e.g. automatic control systems. FPGAs do not include software. Instead, logic functions are programmed onto a chip. FPGAs have been considered to be used in nuclear power plants.

The system that was modelled was an FPGA-based vanadium dynamic signal compensator and trip logic system. Dynamic compensation is needed in the measurement of neutron flux in the reactor core of a nuclear power plant because the detector of the neutron flux has a delay in its response which causes the uncompensated measurement to be incorrect if there is a quick change in the neutron flux.

Radiation-induced failures in FPGAs were included in the model. Ionizing radiation can, for example, cause the state of a storage element to change or a memory cell to get stuck in a state. The following types of failures were included in the model:

- Information corruption in a memory cell
- Corruption of entire data path and loss of system operation
- Pulse through the signal lines and memory elements
- Permanent memory change in memory element
- Permanent logic inversion
- Rupture of the dielectric material gates

The model covered only the FPGA system from the inputs to the output signal. There was no feedback from the controlled process. The top events that were analysed were spurious trip and failure to trip. The model was traced only one time step backwards. For both top events, 48 prime implicants were generated including radiation-induced failures.

### 3.7 Water supply process control

Guarro and Yau built a DFM model for a water supply process control system in [29]. The system has two control modes:

- Normally, pump speed is controlled according to pressure reading in order to maintain the pressure at the set-point.
- When very low flow is detected, pump is set to cycle between on and off states.

The main components of the DFM model are the pump, the controller, the pressuriser and the valve. Their effects on process variables flow rate and pressure are included in the model. The information of the previous states of the components is used in the control logic. Only failures that are modelled seem to be leakage and pump/controller failure.

The model was analysed both to verify the system's correct behaviour in normal conditions and to identify causes for the loss of pressure control. Comprehensive analysis of the results is not presented, only some example prime implicants.

## 4. Conclusions

---

This document presented an overview of DFM and its applications. DFM can be used for the reliability analysis of dynamic systems that contain feedback loops and non-binary variables. A DFM model is a graph representation of a system containing time delays. The same DFM model can be used for the analysis of different top events, including both desired and non-desired end states. The formalism of DFM is simple. It is possible to integrate DFM results into a nuclear power plant PRA model, but it requires some additional consideration and work.

The main application area of DFM has been digital control systems. One reason for this is that the process, control logic, software, physical components and their interdependencies can be included in the same graph model. Dynamic behaviour and causal relationships between physical variables as well as stochastic events can be modelled. Traditional static binary methods, such as fault trees, cannot be used to model dynamic systems with the same accuracy. In addition to component failure combinations causing the top event, it is possible to identify design errors of I&C systems using DFM. In addition to control systems, DFM has benefits in modelling human performance and phased missions.

A drawback of DFM is that complex DFM models are computationally very demanding. The modelling work requires also considerable skills and knowledge on the system. Especially, the modelling of time-dependent behaviour seems to be challenging. Many of the described DFM models did actually not include significant time-dependent modelling, which can be judged based on that the models were traced backwards only one time step.

Most of the described models were quite moderately sized and included less than 50 nodes. Redundancies were not included much. However, it must be noticed that many DFM models not presented in literature exist. There are likely larger models and DFM has probably been applied to some other types of systems as well.

## References

---

- [1] Garrett CJ, Guarro SB, Apostolakis GE. The dynamic flowgraph methodology for assessing the dependability of embedded software systems. *IEEE Transactions on Systems, Man and Cybernetics*. 1995; 25:824-840.
- [2] Al-Dabbagh AW, Lu L. Reliability modeling of networked control systems using dynamic flowgraph methodology. *Reliability Engineering and System Safety*. 2010; 95:1202-1209.
- [3] Yau M, Apostolakis G, Guarro S. The use of prime implicants in dependability analysis of software controlled systems. *Reliability Engineering and Systems Safety*. 1998; 62:23-32.
- [4] Tyrväinen T. Prime implicants in dynamic reliability analysis. *Reliability Engineering and System Safety*. 2016; 146:39-46.
- [5] ASCA Inc. Dymonda. 2010. <http://www.ascainc.com/dymonda/dymonda.html> [Referred 18.3.2016].
- [6] Björkman K. Solving dynamic flowgraph methodology models using binary decision diagrams. *Reliability Engineering and System Safety*. 2013; 111:206-216.
- [7] ASCA Inc. DFM specifications. 2010. [http://www.ascainc.com/dfm/dfm\\_specs.html](http://www.ascainc.com/dfm/dfm_specs.html) [Referred 18.3.2016].
- [8] Tyrväinen T. Risk importance measures in the dynamic flowgraph methodology. *Reliability Engineering and System Safety*. 2013; 118:35-50.
- [9] Karanta I. Implementing dynamic flowgraph methodology models with logic programs. *Journal of Risk and Reliability*. 2013; 227:302-314.
- [10] Houtermans MJM. A method for dynamic process hazard analysis and integrated process safety management [doctoral thesis]. Eindhoven (Netherlands): Technische Universiteit Eindhoven; 2001 May. <http://alexandria.tue.nl/extra2/200111699.pdf>.
- [11] Karanta I. Importance measures for the dynamic flowgraph methodology. Espoo (Finland): VTT Technical Research Centre of Finland, Systems Research; 2011 Dec. Report No. VTT-R-00525-11.
- [12] Tyrväinen T, Björkman K. Modelling common cause failures and computing risk importance measures in the dynamic flowgraph methodology. *Proceedings of the 11th International Probabilistic Safety Assessment and Management*

Conference & The Annual European Safety and Reliability Conference; 2012 Jun 25-29; Helsinki, Finland. Helsinki: The International Association for Probabilistic Safety Assessment and Management (IAPSAM); 2012. 30-Th4-1.

- [13] Aldemir T, Miller DW, Stovsky MP, Kirschenbaum J, Bucci P, Fentiman AW, Mangan LT. Current state of reliability modeling methodologies for digital systems and their acceptance criteria for nuclear power plant assessments. Washington D.C. (USA): U.S. Nuclear regulatory commission, Division of Fuel, Engineering, and Radiological Research; 2006 Feb. NUREG/CR-6901.
- [14] Pinto JMO, Frutuoso e Melo PF, Saldanha PLC. A dynamic failure evaluation of a simplified digital control system of a nuclear power plant pressurizer. Proceedings of the 13th Brazilian Congress of Thermal Sciences and Engineering; 2010 Dec 5-10; Uberlandia, MG, Brazil. Rio de Janeiro: ABCM; 2010.
- [15] Aldemir T, Stovsky MP, Kirschenbaum J, Mandelli D, Bucci P, Mangan LA, Miller DW, Sun X, Ekici E, Guarro S, Yau M, Johnson B, Elks C, Arndt SA. Dynamic reliability modelling of digital instrumentation and control systems for nuclear reactor probabilistic risk assessment. Washington D.C. (USA): U.S. Nuclear regulatory commission, Division of Fuel, Engineering, and Radiological Research; 2007 Oct. NUREG/CR-6942.
- [16] Aldemir T, Guarro S, Mandelli D, Kirschenbaum J, Mangan LA, Bucci P, Yau M, Ekici E, Miller DW, Sun X, Arndt SA. Probabilistic Risk Assessment Modeling of Digital Instrumentation and Control Systems Using Two Dynamic Methodologies. Reliability Engineering and System Safety. 2010; 95:1011-1039.
- [17] Björkman K, Tyrväinen T. Dynamic flowgraph methodology as a part of PRA. Espoo (Finland): VTT Technical Research Centre of Finland; 2015. Report No. VTT-R-04222-14.
- [18] Pinto JMO, Frutuoso e Melo PF, Saldanha PLC. A DFM/Fuzzy/ATHEANA human failure analysis of a digital control system for a pressurizer. Nuclear Technology. 2014; 188:20-33.
- [19] Houtermans M, Apostolakis G, Brombacher A, Karydas D. The dynamic flowgraph methodology as a safety analysis tool: programmable electronic system design and verification. Safety Science. 2002; 40:813-833.
- [20] Karanta I, Maskuniitty M. Reliability of digital control systems in nuclear power plant – modelling the feedwater system. VTT Technical Research Centre of Finland, Systems Research; 2009 May. Report No. VTT-R-01749-08.
- [21] Yau M, Guarro S, Apostolakis G. Demonstration of the dynamic flowgraph methodology using the Titan II space launch vehicle digital flight control system. Reliability Engineering and System Safety. 1995; 49:335-353.
- [22] Yau M, Dixon S, Guarro S. Application of the dynamic flowgraph methodology to the space propulsion system benchmark problem. Proceedings of the 12th International Probabilistic Safety Assessment and Management Conference; 2014 Jun 22-27; Sheraton Waikiki, Honolulu, Hawaii, USA.
- [23] Shi J, Wang G, Tong T. The integrated health monitoring design using the dynamic flowgraph methodology for thermal control systems of payloads. Chemical Engineering Transactions. 2013; 33:211-216.

- [24] Al-Dabbagh AW. Dynamic flowgraph methodology for reliability modelling of networked control systems [master's thesis]. University of Ontario Institute of Technology, Oshawa, 2009.
- [25] Al-Dabbagh AW, Lu L. Dynamic flowgraph modeling of process and control systems of a nuclear-based hydrogen production plant. *International Journal of Hydrogen Energy*. 2010; 35:9569-9580.
- [26] Ahmed F. Probabilistic risk assessment using dynamic flowgraph methodology for copper chloride CANDU-SCWR hydrogen production. *Procedia Computer Science*. 2013; 19:777-785.
- [27] Milici A, Mulvihill R, Guarro S. Extending the dynamic flowgraph methodology (DFM) to model human performance and team effects. Washington D.C. (USA): U.S. Nuclear regulatory commission, Division of System Analysis and Regulatory Effectiveness; 2001 Mar. NUREG/CR-6710.
- [28] McNelles P, Zeng ZC, Renganathan G. Modelling radiation-induced failures in FPGAs using the dynamic flowgraph methodology. *Transactions of the American Nuclear Society*. 2015; 113:415-418.
- [29] Cosgrove J, Guarro S, Romanski G, Yau M. Dynamic modelling and verification of safe-set architectures. Conference proceedings of WESCON/96; 1996 Oct 22-24; Anaheim, CA, USA. Piscataway: The Institute of Electrical and Electronics Engineers, Inc.; 1996. p. 528-533. ISBN 0-7803-3274-1.