Fault Propagation Modelling for Fluid System Health Monitoring

R. REMENYTE-PRESCOTT* and J. D. ANDREWS

Nottingham Transportation Engineering Centre, University of Nottingham, Nottingham, NG7 2RD, UK

Abstract: Fault diagnostics systems are incorporated to determine the health of the system they monitor. There are however times when the diagnostics system reports faults which do not exist. This situation commonly arises at system start-up when high vibration levels exist and the systems are not performing in the same way as when they are operational. Unnecessary shutdowns can occur due to transient behaviour of the system. On autonomous vehicles, such as Unmanned Aerial Vehicles (UAVs), information about the health of the system can be used to support the decision making process and to plan the future system operation. When faults are reported on autonomous systems, where there is no pilot to interpret the conditions reported, a method is needed to establish whether the reported faults do exist. Utilising a fault propagation modelling technique deviations in system variables can be propagated through the system until further evidence of fault presence is observed. If some evidence that contradicts the fault presence is found, the fault can be cancelled and unnecessary shutdowns can be avoided.

In this paper a propagation table method is developed to model fault propagation through a system. The system is broken down into its constituent components and each model shows how process variables depend not only on the state of the component but also on the state of the entire system. The outputs of the two-way fault propagation modelling are values of process variables at different locations in the system. These values can be compared with the symptoms observed and used to cancel or confirm faults. This comparison process is accomplished at each phase that the system goes through during its defined mission. The illustration of the fault propagation methodology is given using an example system, and its application for the fault cancellation process is discussed.

Keywords: *Decision table, propagation table, fault propagation modelling, fault diagnostics.*

1. Introduction

A fault can be defined as an abnormal state of a system, including malfunction of a part, an assembly or the whole system. When a fault occurs, knowing exactly which failure has occurred and caused observed symptoms allows the decision maker to be able to take an appropriate action and reduce consequences of failure. Understanding of symptoms and fault conditions supports the fault diagnostics process, which is essential for the safe operation of the system and maintenance planning. It is beneficial to have a model which takes the system status, indicated by sensors, and predicts possible component failures. There is a considerable interest of this capability for autonomous vehicles, when the fault diagnostics feeds directly into decision making for mission planning. A fault propagation model could also assist in an optimised sensor allocation strategy to maximise fault diagnosability.

During the start-up of the system, diagnosed fault conditions are added to a fault log or report, even when faults occur due to transient behaviour of the system, for example, during the start-up of a fuel pump. Depending on the nature of the faults they can result in system shutdown, especially when the system is operated autonomously and decisions are made according to the fault condition report. Therefore, unnecessary shutdowns of the system could be avoided, when false faults are cancelled after performing the fault propagation modelling. During such a process some means of system representation is needed to allow disturbances induced by faults to be propagated through the system. Introducing deviations in system variables and propagating them through the system could be used to confirm faults, if observations match the expected effects, or cancel faults, if symptoms contradict the expected effects.

In the last three decades many efficient tools have been used to model fault propagation. A number of causal models are used to capture the causal structure of the system and to reason from the qualitative knowledge within the models. The main techniques used in the literature are digraphs [1,2,3], decision tables [4] and fault trees [5,6,7,8]. In recent applications Petri nets are also used to model fault propagation [9].

This paper develops a propagation table method, when (1) no restriction is placed on the number of discrete states used to represent variable deviations, (2) a number of process variables can be modelled using a structure similar to a decision table, and (3) a number of different component failure modes can be considered in the model. Originally in [10] the decision tables described each possible output state as a complete set of combinations of inputs and internal operational or failed states. In the study [4] systems with control loops were modelled using the decision table method. Further developments on the fault propagation modelling using the decision tables were done in [11], when a super component concept was introduced to model sections of a complex diagram as a single component. In [12] failures were defined not only by qualitative descriptions in the tables, but also by quantitative event types, to describe different properties of components with different specifications. A combined decision table and diagraph method was presented in [13,14] to model control loops and circuits in the system. The hybrid approach was based on flexibility of the decision table method but also incorporated a way of analysing control loops, similar to that used in the digraph approach.

Tables, such as decision tables, are suitable for fault propagation modelling, because they describe cause and effect behaviour of the system under normal and failed conditions. The novelty of the propagation table method in this paper is that (1) all process variables are modelled in a single table, (2) component models are built considering not only the states of the component but also the state of the system downstream, and (3) failure effects are recorded not only for the output port but also for the input port of the component which enables a more realistic two-way fault propagation. A system is defined by a schematic diagram, which is decomposed into its constituent components for the modelling. A propagation table is built for each component and the fault propagation algorithm is used to obtain variable deviations in the system. The outcomes can be compared with the observed symptoms and the fault cancellation rules can be applied to confirm or cancel faults. The approach is presented in section 3 and its illustration using a simple system example is given in Section 4.The propagation table can also be used in fault diagnostics, as discussed in section 5.

2. Overview of Fault Propagation Modelling Technique

Fault propagation modelling shows how faults affect process variables and how their deviations distribute through the system. Such an approach can be useful in identifying failures which have caused the system deviations or propagate the deviations of process variables in the system. The latter capability is especially important in the fault cancellation process, which can solve the issue of reported faults which do not exist in reality. Such a situation can arise at system start-up, when due to high vibration levels the system performs differently than in the operational mode. For autonomous vehicles the issue becomes even more important when there is no pilot to interpret the health management reports and ignore false alarms. Therefore, a method to establish fault presence is needed.

Fault propagation modelling is performed using a novel fault propagation table method. There are four steps in the method:

1. System decomposition
2. Development of fault propagation tables
3. Fault propagation algorithm for obtaining process variable values in the system
4. Fault cancellation rule.

Figure 1 shows the basic approach of the technique. A system is modelled as a set of different types of components connected to each other. Each component is represented by a propagation table. The novelty of the method is the ability to perform two-way fault propagation modelling, which outputs values of process variables at different locations in the system. These values show how the system works in the operational mode and what the effects of component failures are. In the latter case variable values can be compared with the symptoms observed and a rule is introduced for cancelling or confirming faults.

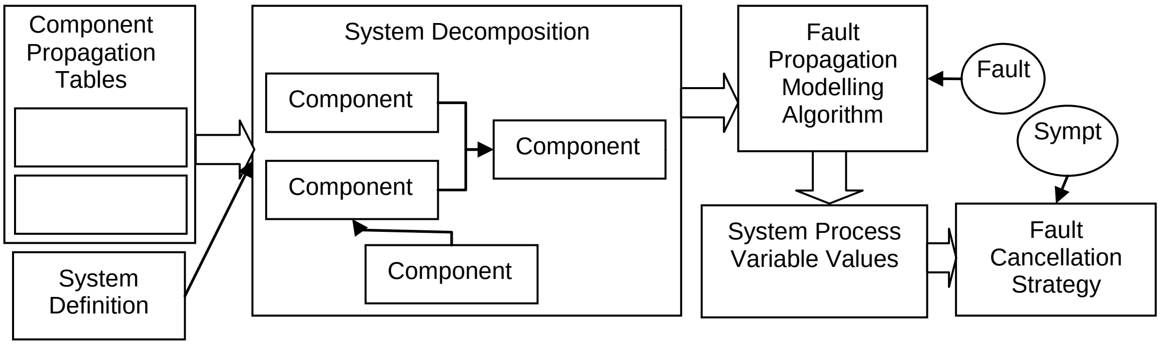
Component Propagation Tables
System Definition
System Decomposition
Component
Component
Component
Component
Fault Propagation Modelling Algorithm
Fault
Sympt
System Process Variable Values
Fault Cancellation Strategy

Figure 1: Structure of the Proposed Approach for Fault Propagation Modelling

3. Fault Propagation Modelling Approach

3.1 System Decomposition

In the component-based approach of system modelling the system is decomposed to its constituent components. Using this approach a schematic diagram is converted into a configuration diagram, where components are represented by rectangles, connected by directed links. Each rectangle contains the name of the component, its unique identification number on the left and the model number on the right. Components are numbered from left to right across the diagram, starting with the process flow and then considering control loops. Types of typical component models in fluid systems are shown in Table 1. The model number is needed for the identification of the component propagation table during the fault propagation.

Rectangles in the configuration diagram are connected by directed links, where the arrows show the direction of the flow or the direction of the information passed around the system. A link is called a location, which has a unique identification number allocated to it. Locations are numbered in a similar manner to the components. In a tabular form of the system diagram each location is described by its upstream and downstream components and their ports, which are used during the fault propagation modelling in order to identify component models upstream and downstream the fault.

In addition to the schematic diagram some structural information about the system is also necessary, *i.e.,* the information about flow loops, control loops, dividers and headers. The presence of these structures affects the general rules of the fault propagation algorithm. The necessary information can be described manually or generated automatically using the criteria, described in this section. Also, it is important to have the information about the flow capacity. In this paper pumps are assumed to work 100%, and if the flow at the input of the divider is 100%, then the flow at one output of the divider is also 100%, if the other output does not contain any flow. If both outputs allow the flow through, the capacity in each output is 50%.

Table 1: Component Models

Model no.	Model description
1	Engine
2	Engine Control
3	Level Control
4	Level Sensor
5	Pipe Single
6	Pipe Divider
7	Pump
8	Pump Control
9	Tank
10	Valve
11	Valve Control

3.2 Component Propagation Tables

In order to model the fault propagation each component is described by a propagation table. This method is suitable for fault propagation modelling, since the table is a type of causal model, when cause and effect behaviour of the system under normal and failed conditions is considered. The main advantage of the propagation table method is its flexibility in terms of entries in the table which is important for the efficient two-way fault propagation modelling technique.

3.2.1 Process Variables and Assumptions

In order to describe component models and their building mechanism, consider the relevant process variables and modelling assumptions. For the schematic diagram four process variables are identified – mass (M), mass flow (F), pressure (P) and level (L). Basic physic laws, governing mass, energy and momentum, are applied in component models. Flow is created by a pressure difference in the system and is modelled together with the pressure. Tank level is a function of flow in and out of the tank. The deviations of these parameters that appear in the tables are considered as follows:

- Mass: present (Y), absent (N).
- Flow/level: zero (Z), lower than expected (L), normal (N) and higher than expected (H).
- Pressure: lower than expected (L), normal (N) and higher than expected (H).

Note that H and L are assumed to be simply high or low to cause a deviation, so no degrees of “highness” or “lowness” are considered in the method.

There are a number of control signals used in the component models, as shown in the models in section 4. For example, a signal to start a pump is defined as Y_{start}. If during the system operation the control signal stays the same, it is described as a lack of the signal to perform an opposite operation, for example, a lack of signal to shut the pump, N_{shut}. If the control signal changes to cause the opposite effect, it is described as a signal to perform an opposite action, for example, a signal to shut the pump, Y_{shut}.

A number of modelling assumptions are set:

- System pressure is higher than the atmospheric pressure outside the system
- Flow through a pipe leak is smaller than flow in the system
- There is no reverse flow in the system
- The mass is absent if it cannot be maintained during the operation of the component
- A component leakage creates local system effects only; therefore, fault effects are propagated to neighbouring components only.

3.2.2 Propagation Table Structure

The propagation table method developed in this paper has a number of advantages from the previous applications of decision tables. First of all, instead of having a number of different tables for the same component but different variable, the method can accommodate deviations of mass, flow, pressure and level in the same table. Since these parameters are related through the laws of physics it is beneficial to store their deviations in one table. Secondly, the propagation table accommodates not only every component failure mode but also the state of other components downstream. For example, zero flow out of the pipe can be caused not only by the pipe blockage but also by other blockages downstream of the pipe. Finally, not only the outputs but also the inputs are affected by faults; therefore, each input and output is modelled in the table. These advantages of the new method are beneficial in performing two-way fault propagation in fluid systems.

There are four types of columns in the propagation table, which are described as follows:

1. **Inputs:** A set of relevant process variables at all in ports, out ports and control signals, if appropriate, are considered in the input column. In order to reduce the table size the pressure and flow values are omitted from component models with a single in port. For the modelling it is assumed that these values are P and F for pressure and flow respectively.
2. **Component state:** Working and failed states of the component are considered in this column. For example, in the pipe model working, leaking, ruptured or blocked states are considered.
3. **Conditions downstream:** These conditions describe the state of the system downstream. The flow system can be working, blocked or ruptured.
4. **Outputs:** These conditions are described by a set of relevant process variables at all in and out ports of the component, and control signals, if

appropriate. For example, in the pipe model the mass, flow and pressure into the pipe and out of the pipe are considered.

Once component failure modes are chosen, the propagation table is filled taking into account relevant inputs, combining them with each state of the component and relevant conditions downstream and obtaining the outputs. For example, due to a pipe blockage or a blockage downstream, there is no flow into the pipe and out of the pipe, and pressure increases before the blockage, as shown in Table 2, rows 2 and 4. Note that not every model has all columns; the presence depends on their relevance to the behaviour of the component, as seen in the component models in section 4. Finally, a reduction of the table is performed by introducing more general entries and using the “-” symbol. The “-“ notation indicates the “don’t care” condition of the entry, which means that a specified set of entries in a row will occur regardless of the entry. For example, if the pipe is blocked there will be the same effects regardless of the conditions downstream.

Table 2: A Propagation Table for a Pipe

No	Inputs	Component state	Conditions downstream	Outputs					
	M			P_{in}	F_{in}	M_{in}	P_{out}	F_{out}	M_{out}
1	Y	W	W	P1	F	Y	P2	F	Y
2	Y	W	B	≥P1	Z	Y	≥P2	Z	Y
3	Y	W	R	≤P1	≥F	Y	≤P2	≥F	Y
4	Y	B	-	≥P1	Z	Y	≤P2	Z	N
5	Y	R	-	≤P1	≥F	Y	≤P2	Z	N
6	Y	L	W	≤P1	≥F	Y	≤P2	≤F	Y
7	Y	L	B	≥P1	≤F	Y	≤P2	Z	Y
8	Y	L	R	≤P1	≥F	Y	≤P2	≥F	Y
9	N	-	-	≤P	Z	N	≤P	Z	N

3.2.3 Formalization of the System Operation

A table with expected component states during the normal behaviour of the system in each phase is built, which is used to decide whether the component state is faulty or working. For example, if an engine feed valve is controlled to be closed in phase I and open in phase II, the valve failure closed in phase I does not affect the normal system behaviour, since the valve is expected to be closed in phase I. Fault effects become important in phase II, when the valve is controlled to open, therefore, the fault propagation modelling is performed in phase II.

As it was mentioned in the step of system decomposition, there are situations when the general rules of modelling are not sufficient to describe the algorithm and some structural information about the system is needed. The following definitions are used in the modelling algorithm to formalise the structural information of the system:

1. **Flow loop.** A loop is a part of the system when the trace in the direction of the flow passes over the same component twice. Loops appear due to alternative lines of flow in the system. They can be obtained manually or using a simple tracing algorithm.
2. **Control loops.** A control loop usually has a sensor, a controller and a valve. In this paper a feedback control system is modelled, when the sensor monitors the output, feeds the data to the controller which continuously adjusts the control input. For example, a feedback tank level control loop consists of the level sensor which monitors the level, the level controller, which adjusts the operation of the pump to obtain appropriate flow from the tank, and the pump, which has similar effects to those of a valve. The configuration of the loop is given by a set of locations in the direction of the

information flow which can be obtained manually or using an algorithm of the control loop definition.

3. **Pipe divider.** A pipe divider has a single input and a number of outputs, which are given by a set of locations starting with a single in port and two out ports.
4. **Pipe header.** A pipe header has a number of inputs and a single output, which is given by a set of locations in a similar manner to the pipe divider.
5. **Gap components.** In schematic diagrams there are components that break the continuity of flow. This happens when the component has a capacitance effect, for example, a tank. Also, dividers and headers are called gap components, since multiple inputs or outputs also disturb the continuity of flow.

3.2.4 Modelling Algorithm

The purpose of the fault propagation modelling is to record values of process variables at each location in the system. These values are obtained from relevant rows in the component models. The process starts at a set location, recording the parameter values in and out of the component from the output column in the model. Then these values are passed to the component models downstream and upstream the system to record parameter values at other locations. The function, shown in the flowchart in Figure 2, recursively calls itself until all the locations are visited and parameter values are recorded.

There are four main steps in the algorithm:

1. Start the modelling at a set location
2. Propagate downstream the system, incorporating control loops, flow loops and dividers
3. Propagate upstream the system, incorporating control loops, flow loops and dividers
4. Calculate tank level.

The modelling is performed for each phase of the system operation, and faults and their effects are carried over from one phase to another. For example, if the tank is leaking in phase I and the level drops to low, at the beginning of phase II the tank level is also low. Also, if the failure mode does not violate the expected state of the component, the system is operating normally, even when a fault has occurred. However, the failure mode can cause failure effects in the future phases.

3.2.4.1 Start of the Modelling Process

The modelling process starts at a set location in the system. The set location is the source of the system material. Since the scope of this paper is fluid systems, control signals are related with the flow in the system and its control, and the source of fluid is a tank. The flow is present in the system if a relevant pump is working and the relevant valves are open. Therefore, in fluid systems the modelling starts initiating the flow out of the tank in the part of the system with a working pump and an open valve. When a fault occurs, the set location for the start of the process is the failed component.

The propagation table of the component at the set location is identified according to the model number in Table 1 and the component model. A row that corresponds to the component state, appropriate input values, control signal and working conditions downstream are considered. Parameter values from the output column in that row are recorded at the locations of in and out ports of the component.

3.2.4.2 Propagation Downstream the System

Parameter values at the out port location of the component become the values at the in port of the component downstream and are used to find a suitable row in the model. Following exceptions are made:

- If the component is a part of the flow loop, each component downstream is considered until the start of the loop is reached. This is also the case for the control loop. In addition to reaching the end of the loop, the modelling of the output, *i.e.*, the control signal, is also performed.
- If the component is a divider, the out ports are considered one-by-one. In order to choose an appropriate row in the divider model, the line is considered in the open state, if a valve on the line is open and there are no obstacles for the flow. If the valve is closed or there are some other obstacles for the flow, the line is considered blocked.

The process continues until the end of the system, the end of the loop or a gap unit is reached.

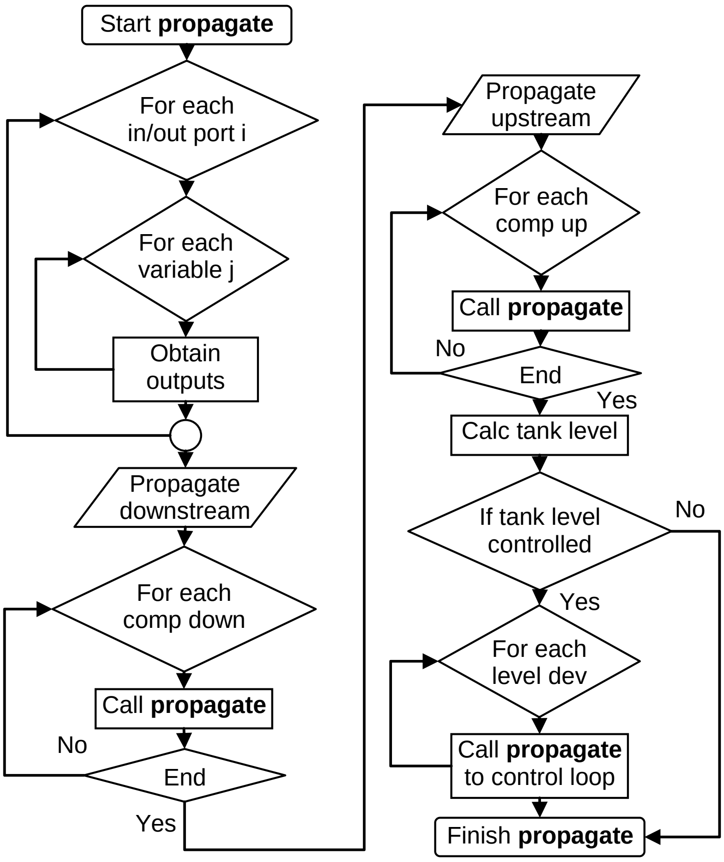

Figure 2: The Flowchart of the Algorithm

3.2.4.3 Propagation Upstream the System

Considering the component upstream, parameter values at the location of the in port to the component are recorded. During this step of the algorithm the state of the failed component becomes the state of conditions downstream. A list of fault categories is shown Table 3, where faults which cause the same effects are in the same category.

Note, that a leakage is considered in the category of working states, since the assumption was made to consider the local effects of the leakage only. Also, the pump failure off/N_{on} does not fit in any category of faults. This fault results in the absence of the mass and the parameter values are propagated through the system. Following exceptions are made:

- If the component is a part of the flow loop, each component upstream is considered until the start of the loop is reached. If the component is a part of the control loop, no upstream propagation is performed, since only one way propagation, *i.e.,* downstream of the component, is possible in the component models of the control loop. Changes in the tank level are calculated in Step 4.
- If the component is a divider, all the out ports are considered first before going upstream to the in port. Some out ports might need to be revisited, if parameter values are affected by the divider model. Also, for components upstream the divider, the value of conditions downstream might change from a blockage to working conditions, for example, if a blockage can be rectified using a reconfiguration option, as described in the example system in section 4.

The process continues until the start of the system, the start of the loop or a gap unit is reached.

Table 3: Categories of Component Faults

No	Fault	Category
1	Pipe/tank blockage	Blockage
2	Pipe/tank rupture	Rupture
3	Pipe/tank leak	Working
4	Valve fails open/N_{close}	Fails open
5	Valve fails closed/N_{open}	Blockage
6	Engine fails off /N_{on}	Blockage

3.2.4.4 Tank Level Calculation

The tank level is calculated once all the locations downstream the tank are visited. It is calculated as a sum of all the positive flows in and of all the negative flows out:

$$\text{Level} = \text{Level}_{\text{init}} + \sum_{i} F_{in} - \sum_{i} F_{out} \qquad (1)$$

Initially the level in the tank is normal, *i.e.*, $\text{Level}_{\text{init}}$ = N. If $\sum_i F_{in}$ exceeds $\sum_i F_{out}$, the level is high, if $\sum_i F_{out}$ exceeds $\sum_i F_{in}$ the level is low, otherwise, the level is normal.

If the level is controlled, the value of the tank level is passed to the control loop. If the value of the tank level changes, it is sent to the control loop and the process continues until all the relevant level deviations are considered, as shown in Figure 2.

The steps of the algorithm are applied until all the locations are visited.

3.2.5 Multiple Fault Propagation Case

The previously described algorithm considers the single fault propagation process only. However, it can be carried over to the multiple fault propagation case. Multiple faults are considered in the order that they occur during the operation. If they occur at the same time, a logical approach is to consider them in the order that components are positioned in the system in the direction of flow. If one of the faults is a pump failure off, this fault is considered first, even if there are some faults upstream the pump. This is due to the fact that the pump failure terminates the flow in the system and effects of other failures do not need to be modelled. In such a case unnecessary modelling steps can be avoided.

3.2.6 Automated Approach

The main advantage of the novel method is two-way fault propagation modelling, which corresponds well to real system behaviour. The method can be used to model complex fluid systems that contain flow and control loops, have different and reconfigurable behaviour in each phase. Due to the flexibility of the method, *i.e.*, an unrestricted number of entries in the table, the size of the models can become large, but an automated modelling approach will ease the implementation of the algorithm. Some expert knowledge is needed to construct component models. However, once models are obtained they can be reused for each type of the component while recursively following the straightforward steps of the algorithm. Also, the decomposition can be error-prone, therefore, the conversion approach could also be implemented on the computer to manage this time-consuming process.

3.3 Fault Cancellation Strategy

The fault propagation modelling technique can be used to confirm or cancel faults which are added to a fault condition report during the start-up of the system. If the system is operated autonomously and decisions are made according to the fault condition report, such faults can cause unnecessary shutdowns of the system.

A following fault cancellation rule is described:

Using the fault propagation modelling technique deviations that arise from the failure listed in the fault condition report are propagated through the system looking for further evidence of the failure. If the evidence of the successful operation of the component is observed in the later phases of the operation, the failure can be cancelled due to the contradiction between symptoms and expected effects. If there is no evidence of the successful operation of the component, the failure is confirmed and related actions should be taken. For example, in the operation of the autonomous aircraft the fault cancellation can occur if the evidence against the faults recorded during the start-up is known before the take-off phase. The aircraft can progress through all phases up to this point. If such evidence is not found, the fault is confirmed and the future flight might need to be cancelled.

The strategy is illustrated using the example system in the following section.

4. Illustration of the Method

4.1 Example System Definition

Consider an example system represented by the schematic diagram, shown in Figure 3. This simple system contains components of the main equipment classes and is used to illustrate the fault propagation modelling technique using the propagation table method.

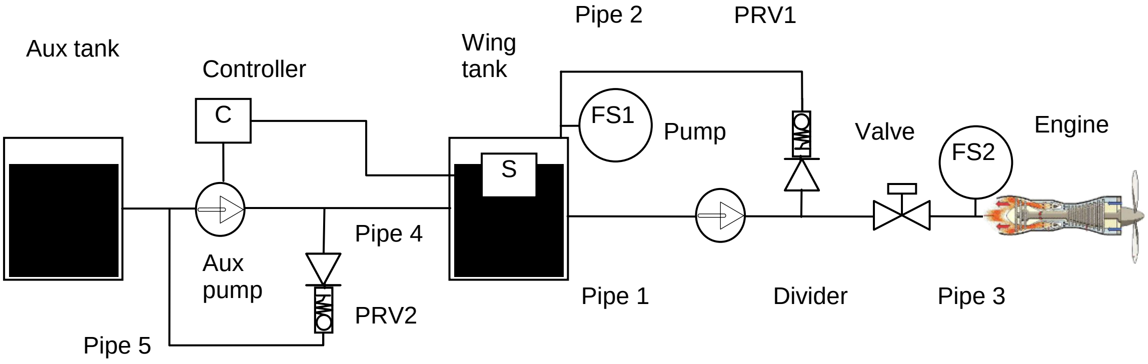

Figure 3: A Schematic Diagram of a Simple System

The aim of the system is to feed fuel from the wing tank to the engine. The system operation is considered in two phases:

Phase I:

In the start-up phase the engine pump is commanded to start pumping. The engine feed valve is closed; therefore, the flow is directed back to the tank via the pressure relief valve, which stabilises the pressure and flow during the start-up. The flow sensor FS1 monitors the flow back to the tank. The auxiliary transfer pump cannot be operated in this phase; the level control loop is inactive.

Phase II:

In the normal operation phase the engine feed valve is commanded to open and let the flow to the engine, monitored by FS2. As soon as the level in the wing tank drops below the expected level due to the fuel burnt by the engine, a signal is sent to a controller which then activates the auxiliary transfer pump and starts taking the fuel from the auxiliary tank. In normal conditions there is no flow back to the wing tank. In the case of a blockage downstream the engine pump, the pressure relief valve PRV1 opens and allows the flow back to the wing tank, decreasing the pressure and avoiding pipe rupture. Similarly, PRV2 directs the flow back to the auxiliary tank, if the flow cannot reach the wing tank.

Following initial conditions are defined:

- Before the system operation starts the tank is filled up to a normal level.
- A constant demand is placed on the engine feed pump.
- Fuel is present in the lines before the pump is turned on.

4.2 System Decomposition

During the decomposition the schematic diagram in Figure 3 is replaced by the corresponding configuration diagram, shown in Figure 4. Each component is allocated a rectangle, for example, the wing tank is allocated the identification number 7 and the model number is 9, *i.e.*, a tank model in Table 1.

Table 4: A Tabular Configuration of the Example System

Location no.	Upstream		Downstream	
	Component	Port	Component	Port
1	1	out	2	in
2	2	out	3	in
3	3	out1	4	in
4	4	out	5	in
5	5	out	1	in
6	3	out2	6	in
7	6	out	7	in
8	7	out	8	in
9	8	out	9	in
10	9	out	10	in
11	10	out1	11	in
12	11	out	12	in
13	12	out	7	in
14	10	out2	13	in
15	13	out	14	in
16	14	out	15	in
17	7	level	16	in
18	16	out	17	in
19	17	control	2	control

20	18	control	9	control
21	19	control	13	control
22	20	control	15	control

A tabular form of the system diagram is shown in Table 4, for example, at location 3 the upstream component is the divider, which is connected to location 3 via the port out1, and the downstream component is pressure relief valve PRV2, which is connected to it via the port in.

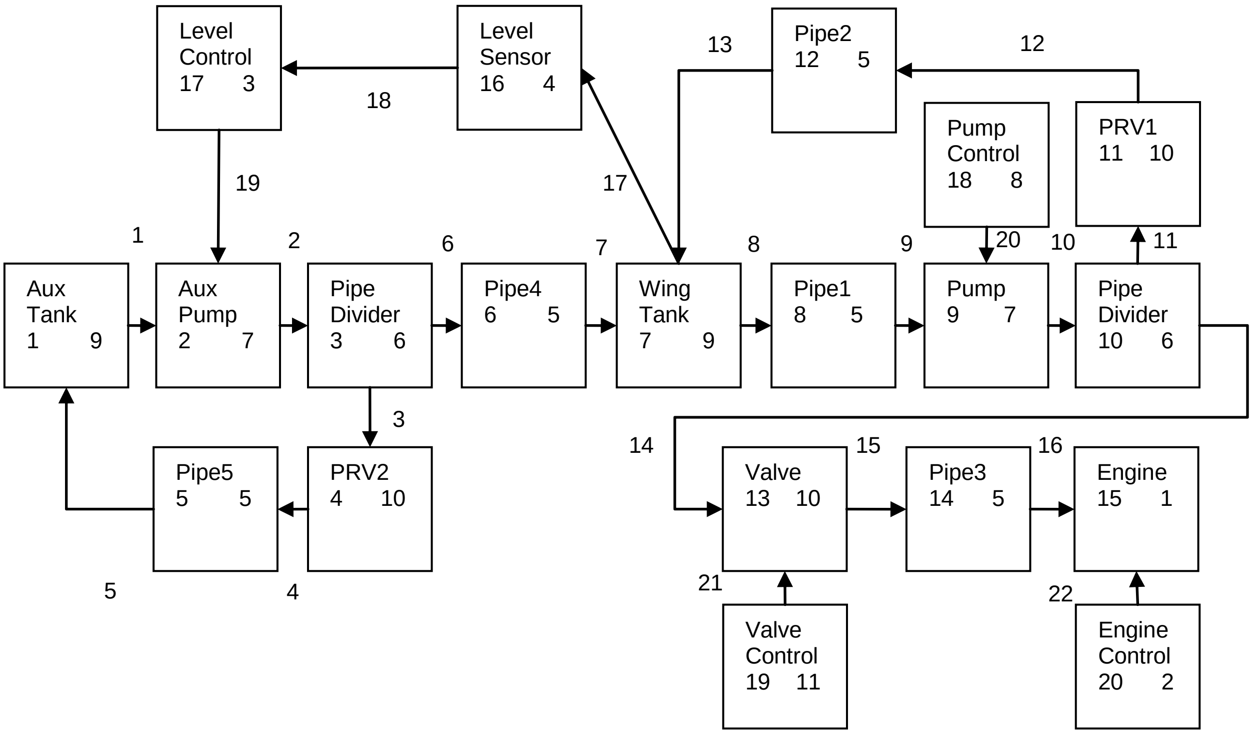

Figure 4: A Configuration Diagram of the Example System

1.1 Propagation Tables

The propagation tables for the main types of components in fluid systems are given in this section.

Table 5: A Propagation Table for a Level Controller

No	Inputs	Component state	Outputs
	Level		Control signal
1	L	W	Y_{start} (N_{shut})
2	≥N	W	N_{start} (Y_{shut})
3	-	CFH	N_{start} (Y_{shut})
4	-	CFL	Y_{start} (N_{shut})
5	-	Inactive	N_{start} (Y_{shut})

Table 6: A Propagation Table for a Level Sensor

No	Component state	Outputs
		Level
1	W	Level
2	SFH	H
3	SFL	L

Table 7: A Propagation Table for a Pipe Divider

No	In	Cond Down out1	Cond Down out2	Outputs								
	M			P_{in}	F_{in}	M_{in}	P_{out1}	F_{out1}	M_{out1}	P_{out2}	F_{out2}	M_{out2}

1	Y	B	B	≥P	Z	Y	≥P	Z	N	≥P	Z	N
2	Y	B/O	R	≤P	≥F	Y	≤P	Z	N	≤P	≥F	Y
3	Y	B	O	P	F	Y	P	Z	N	P	F	Y
4	Y	R	B/O	≤P	≥F	Y	≤P	≥F	Y	≤P	Z	N
5	Y	R	R	≤P	≥F	Y	≤P	≥F	Y	≤P	≥F	Y
6	Y	O	B	P	F	Y	P	F	Y	P	Z	N
7	Y	O	O	P	F	Y	P	≤F	Y	P	≤F	Y
8	N	-	-	≤P	Z	N	≤P	Z	N	≤P	Z	N

Table 8: A Propagation Table for a Pump

No	Inputs		Component State	Conditions Downstream	Outputs					
	M	Control Signal			P_{in}	F_{in}	M_{in}	P_{out}	F_{out}	M_{out}
1	Y	Y_{start} (N_{shut})	W/ON	W	P1	F	Y	P2	F	Y
2	Y	-	F/ON	W	P1	F	Y	P2	F	Y
3	Y	N_{start} (Y_{shut})	W/OFF	-	≤P1	Z	N	≤P1	Z	N
4	Y	-	F/OFF	-	≤P1	Z	N	≤P1	Z	N
5	Y	Y_{start} (N_{shut})	W/ON	B	P1	F	Y	≥P2	Z	Y
6	Y	-	F/ON	B	P1	F	Y	≥P2	Z	Y
7	Y	Y_{start} (N_{shut})	W/ON	R	P1	F	Y	≤P2	F	Y
8	Y	-	F/ON	R	P1	F	Y	≤P2	F	Y
9	N	-	-	-	≤P1	Z	N	≤P1	Z	N

Table 9: A Propagation Table for a Tank

No	Inputs		Component State	Conditions Downstream	Outputs			
	M	Level			P_{out}	F_{out}	M_{out}	F_{loss}
1	Y	N	W	W	P	F	Y	Z
2	Y	L	W	W	≤P	≤F	Y	Z
3	Y	-	W	B	≥P	Z	Y	Z
4	Y	-	W	R	≤P	≥F	Y	Z
5	Y	-	B	-	≤P	Z	N	Z
6	Y	-	R	-	≤P	Z	N	H
7	Y	-	L	W	P	≤F	Y	L
8	Y	-	L	B	P	Z	Y	L
9	Y	-	L	R	≤P	≥F	Y	L
10	N	-	-	-	≤P	Z	N	Z

Table 10: A Propagation Table for a Valve

No	Inputs		Comp State	Cond Down	Outputs					
	M	Control Signal			P_{in}	F_{in}	M_{in}	P_{out}	F_{out}	M_{out}
1	Y	Y_{open} (N_{close})	W/O	W	P1	F	Y	P2	F	Y
2	Y	-	F/O	W	P1	F	Y	P2	F	Y
3	Y	N_{open} (Y_{close})	W/C	-	≥P1	Z	Y	≤P2	Z	N
4	Y	-	F/C	-	≥P1	Z	Y	≤P2	Z	N
5	Y	Y_{open} (N_{close})	W/O	B	≥P1	Z	Y	≥P2	Z	Y
6	Y	-	F/O	B	≥P1	Z	Y	≥P2	Z	Y
7	Y	Y_{open} (N_{close})	W/O	R	≤P1	≥F	Y	≤P2	≥F	Y
8	Y	-	F/O	R	≤P1	≥F	Y	≤P2	≥F	Y
9	N	-	-	-	≤P1	Z	N	≤P1	Z	N

1.2 Fault Propagation Modelling Algorithm

1.2.1 Formalisation of the System Operation

Table 11 shows what component states are expected in each phase during the normal behaviour of the system.

Table 11: Expected Working States for System Components

Comp id	Phase I	Phase II	Reconf PRV1	Reconf PRV2
1	W	W	-	-
2	W/OFF	W/ON	-	-
3	-	-	-	-
4	W/C	W/C	-	W/O
5	W	W	-	W
6	W	W	-	-
7	W	W	-	-
8	W	W	-	-
9	W/ON	W/ON	-	-
10	-	-	-	-
11	W/O	W/C	W/O	-
12	W	W	W	-
13	W/C	W/O	-	-
14	W	W	-	-
15	W	W	-	-
16	W	W	-	-
17	Inactive ≥ N_{start}	W	-	-
18	W ≥ Y_{start} (N_{shut})	W ≥ N_{shut}	-	-
19	W ≥ N_{open}	W ≥ Y_{open} (N_{close})	-	-
20	W ≥ N_{on}	W ≥ Y_{on} (N_{off})		-

Note, that in addition to phase I and II, two reconfiguration options in phase II are also described. For example, in order for the reconfiguration PRV1 to occur successfully, *i.e.*, the flow is redirected back to the wing tank due to blockages downstream the pump, the pressure relief valve PRV1 has to open and the pipe downstream has to work. These conditions are modelled if the reconfiguration PRV1 is present. The rest of the component states in the reconfiguration column are left unidentified, since it is impractical to list all possible combinations of component states, when the reconfiguration occurs.

Control signals are also used in the modelling, as shown in the entries of the control components in rows from 17 to 20 in Table 11. For example, in phase I the engine pump controller, component 18, is expected to be in the working state and send the signal to start the pump, *i.e.*, Y_{start}. Flow and control loops are also identified. For example, the flow line back to the auxiliary tank through PRV2 forms a flow loop, *i.e.*, 1-2-3-4-5-1, and the tank level control loop is represented by 17-18-19.

4.2.1 Modelling Algorithm

4.2.1.1 Start of the Modelling Process

The operation starts in phase I when the engine pump starts and the engine valve is closed. The set location for the modelling of the example system is the wing tank. According to the control signals in Table 11 in phase I the engine feed pump starts taking the fuel from the wing tank. The tank model (model Table 9–T9) and row 1(R1) is considered, since in addition to the mass present (the initial assumption), the level in the tank is normal, the tank and the system downstream is working. This row gives the parameter values at location 8, *i.e.,* at the out port of the tank, P_8 = P, F_8 = F, M_8 = Y, F_{loss} = Z.

Alternatively, consider a failure scenario. For example, if the fuel feed pump failure is considered in phase II, then the fault propagation in phase II starts with the failed component model, *i.e.*, the pump model. Since the pump fails off, T7R4 (model table 7, row 4) is used to obtain deviations in and out of the pump, *i.e.*, at locations 9 and 10, $P_9 \leq P$, $F_9 = Z$ and $M_9 = N$, $P_{10} \leq P$, $F_{10} = Z$ and $M_{10} = N$.

4.2.1.2 Propagation Downstream the System

During the propagation downstream, the values at the out port location of the component become the values at the in port of the component downstream.

In the normal operation, according to Table 4 the downstream component of the wing tank is pipe 1. Considering T5R1, *i.e.*, mass is present, the pipe is working and the conditions downstream are working, gives $P_9 = P$, $F_9 = F$ and $M_9 = Y$. The values at other locations are obtained in a similar way until the end of the system is reached.

For the pump failure scenario in phase II, the values at location 10 are passed downstream until the end of the system is reached.

4.2.1.3 Propagation Upstream the System

During this step, parameter values at the in port of the component are recorded, when the state of the failed component becomes the state of conditions downstream. In the normal operation, there are no components upstream the wing tank, since the control loop is considered separately. Therefore, the algorithm moves to the step of tank calculation.

In the pump failure situation, the pipe model for the upstream component pipe 1 is considered, and the failure is treated as the absence of the mass. Therefore, T5R9 gives parameter values at location 8, *i.e.,* P8 ≤ P, F8 = Z and M8 = N. The process stops because the gap unit, *i.e.*, the tank, is reached.

4.2.1.4 Tank Level Calculation

The tank level is calculated using equation 1, once all the locations downstream the tank are visited. In the normal operation, the level is normal, *i.e.,* Level = N + F_{back} - F_{out} = N, ($F_{back} = F_{out}$). It is also the case for the pump failure scenario in phase II, *i.e.,* Level = N + F_{in} - F_{out} = N, ($F_{in} = F_{out} = Z$). This value is sent to the control loop and the process continues until all the relevant level deviations are considered, as shown in Figure 2. The algorithm and the parameter values for the pump failure scenario are shown in Table 12.

Table 12: Pump Failure Propagation in Phase II

Stage	Model	Loc	Parameter values
Start of the Modelling	T7R4	9	$P_9 \leq P$, $F_9 = Z$, $M_9 = N$
		10	$P_{10} \leq P$, $F_{10} = Z$, $M_{10} = N$
Downstream the System	T6R8	11	$P_{11} \leq P$, $F_{11} = Z$, $M_{11} = N$
		14	$P_{14} \leq P$, $F_{14} = Z$, $M_{14} = N$
	T10R9	12	$P_{12} \leq P$, $F_{12} = Z$, $M_{12} = N$
	T5R9	13	$P_{13} \leq P$, $F_{13} = Z$, $M_{13} = N$
	T10R9	15	$P_{15} \leq P$, $F_{15} = Z$, $M_{15} = N$
End of System	T5R9	16	$P_{16} \leq P$, $\mathbf{F_{16} = Z}$, $M_{16} = N$
Upstream the System	T5R9	8	$P_8 \leq P$, $F_8 = Z$, $M_8 = N$
Tank level Calculation	Tank Level	17	$Level_{17}$ = N + F_{in} - F_{out} = N ($F_{in} = Z$, $F_{out} = Z$)
Control Loop	T4R1	18	$Level_{18}$ = N
	T3R2	19	N_{start}

Downstream the System	T9R10	1	$P_1 \leq P$, $F_1 = Z$, $M_1 = N$, $F_{loss} = Z$
	T7R9	2	$P_2 \leq P$, $F_2 = Z$, $M_2 = N$
	T6R8	3	$P_3 \leq P$, $F_3 = Z$, $M_3 = N$
		6	$P_6 \leq P$, $F_6 = Z$, $M_6 = N$
	T10R9	4	$P_4 \leq P$, $F_4 = Z$, $M_4 = N$
	T5R9	5	$P_5 \leq P$, $F_5 = Z$, $M_5 = N$
	T5R9	7	$P_7 \leq P$, $F_7 = Z$, $M_7 = N$
Tank Level Calculation	Tank Level	17	$Level_{17} = N$

4.3 Fault Cancellation Example

The fault cancellation strategy is illustrated using an example of transient behaviour during the start-up of the fuel pump. If the pump failure appears in the fault condition report, the system might experience unnecessary shutdown. Assume the following readings of the two flow sensors in phase I and II, shown in Table 13:

Table 13: Sensor Readings

Phase	FS1	FS2
I	$F_{13} = Z$	$F_{16} = Z$
II	$F_{13} = Z$	$F_{16} = N$

According to the readings in phase I, no flow at location 13 is an unexpected measurement, since in phase I the flow is directed back to the wing tank, *i.e.,* via location 13. Therefore, a list of potential failures that could have caused this deviation is produced, for example, P2B, PRV1B, PumpF/OFF, etc. This list can be obtained using some diagnostics technique. Assume that the pump failure is recorded in the fault condition report. According to the fault cancellation rules further evidence of this failure is needed to confirm the fault. The deviations in system variables due to this failure are propagated through the system using the propagation tables, as seen in Table 12. The pressure drops and there is no flow and no mass in the system.

If the pump is failed, as concluded from the FS1 reading in phase I, in phase II there should be no flow at location 16, *i.e.*, no flow to the engine, $F_{16} = Z$. This value contradicts the FS2 reading in phase II, *i.e.,* $F_{16} = N$. Due to this evidence it can be concluded that the no flow reading at location 13 in phase I was not caused by the pump failure, and therefore, the pump failure can be cancelled. $F_{13} = Z$ was caused by the transient behaviour of the pump. If there are more faults in the fault condition report, they are considered in a similar way.

5. Fault Diagnostics using the Propagation Table Method

In addition to the fault cancellation strategy, the propagation table method presented in this paper can also be used in fault diagnostics. The main principle of the fault diagnostics technique is to map symptoms of possible failure modes in component models until all locations in the diagram are visited. Downstream and upstream components from the location of the symptom are considered according to the rules similar to the fault propagation rules. If a tank level measurement is known, all possible combinations of the flow in and out of the tank, that could have caused the tank level deviation, are considered and their causes are listed. If multiple measurements are known, the most upstream measurement is considered first. Only failure modes that satisfy all the measurements are considered in the list of faults. Non-minimal combinations are always removed.

6. Conclusions

A fault propagation modelling technique has been developed employing a novel propagation table method. The methodology takes into account the schematic diagram of the system and the description of the system behaviour in each phase. The propagation table method accommodates an unlimited number of component and system states and allows a two-way modelling process. Component models are used to propagate the deviations of process variables through the system until all the locations in the diagram are visited. Expected outcomes are compared with observed symptoms and can be used to confirm the fault, if the outcome agrees with the symptom, or to cancel the fault, if a contradiction is observed. The suitability of the method to model complex systems is also discussed.

Acknowledgement: John Andrews is the Royal Academy of Engineering and Network Rail Professor of Infrastructure Asset Management. He is also Director of The Lloyd's Register Educational Trust Centre* for Risk and Reliability Engineering at the University of Nottingham. Rasa Remenyte-Prescott is The Lloyd's Register Educational Trust Lecturer in Risk and Reliability Engineering at the University of Nottingham. Both authors gratefully acknowledge the support of the appropriate organisations.

*The Lloyd's Register Educational Trust is an independent charity working to achieve advances in transportation, science, engineering and technology education, training and research worldwide for the benefit of all.

References

[1]. Lapp, S.A. and J.G. Powers. *Computer-aided Synthesis of Fault Trees*. IEEE Transactions on Reliability 1977; 26(1): 2-13.

[2]. Andrews, J.D. and E.G. Brennan. *Application of the Digraph Method of Fault Tree Construction to a Complex Control Configuration*. Reliability Engineering and System Safety 1990; 28(3): 357-384.

[3]. Kelly, E.M. and L.M. Bartlett. *Enhanced Diagnosis of Faults Using the Digraph Approach Applied to a Dynamic Aircraft Fuel System*. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability 2008; 222(4): 561-572.

[4]. Kumamoto, H. and E.J. Henley. *Safety and Reliability Synthesis of Systems with Control Loops*. American Institute of Chemical Engineering Journal 1979; 200(2): 108-113.

[5]. Kelly, B.E. and F.P. Lees. *The Propagation of Faults in Process Plants: 1. Modelling of Fault Propagation*. Reliability Engineering 1986; 16(1): 3-38.

[6]. Assaf, T. and J.B. Dugan. *Automatic Generation of Diagnostic Expert Systems from Fault Trees*. Proceedings of Annual Reliability and Maintainability Symposium 2003. 143-147.

[7]. Assaf, T. and J.B. Dugan. *Diagnostic Expert Systems from Dynamic Fault Trees*. Proceedings of Annual Reliability and Maintainability Symposium 2004, 444-450.

[8]. Hurdle, E.E., L.M. Bartlett and J.D. Andrews. *Fault Diagnostics of Dynamic System Operation using a Fault Tree Based Method*. Reliability Engineering & System Safety 2009; 94(9): 1371-1380.

[9]. Wang, Y.F., J.Y. Wu and C.T. Chang. *Automatic Hazard Analysis of Batch Operations with Petri Nets*. Reliability Engineering & System Safety 2002; 76(1): 91-104.

[10]. Salem, S.L., G.E. Apostolakis and D. Okrent. *A New Methodology for the Computer-aided Construction of Fault Trees*. Annals of Nuclear Energy 1977; 4(9-10): 417-433.

[11]. Han, S.H., T.W. Kim, Y. Choi and K.J. Yoo. *Development of a Computer Code AFTC for Fault Tree Construction Using Decision Table and Super Component Concept*. Reliability Engineering & System Safety 1989; 25(1): 15-31.

[12]. Wang, J.D. and T.S. Liu. *A Component Behavioural Model for Automatic Fault Tree Construction*. Reliability Engineering & System Safety 1993; 42(1): 87-100.

[13]. Andrews, J.D. and J.J. Henry. *A Computerized Fault Tree Construction Methodology*. Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering 1997; 211(3): 171-183.

[14]. Andrews, J.D. and J.J. Henry. *Computerized Fault Tree Construction for a Train Braking System,* Quality and Reliability Engineering International 1997, 13(5): 299-309.

Rasa Remenyte-Prescott is The Lloyd's Register Educational Trust Lecturer in Risk and Reliability Engineering in Nottingham Transportation Engineering Centre (NTEC) at the University of Nottingham. Rasa gained BSc and MSc degrees with honour in mathematics from Kaunas University of Technology, Lithuania. Following this Rasa undertook her Doctorate research at Loughborough University on systems reliability modelling of non-coherent systems using the Binary Decision Diagram technique. Rasa's current research interests involve fault diagnostics techniques and network infrastructure asset management methods.

John D. Andrews is the Royal Academy of Engineering Professor of Infrastructure Asset Management in the Nottingham Transportation Engineering Centre (NTEC) at the University of Nottingham. Prior to this he worked in the Department of Aeronautical and Automotive Engineering at Loughborough University, UK. The prime focus of his research has been on methods for predicting system reliability in terms of the component failure probabilities and a representation of the system structure. Much of this work has concentrated on the Fault Tree technique and the use of the Binary Decision Diagrams (BDDs) as an efficient and accurate solution method. He is the author of around 200 research papers on this topic and is joint author, along with Bob Moss, of a text book, Reliability and Risk Assessment, now in its second edition, published by ASME. John is a member of the Safety and Reliability group of the Institution of Mechanical Engineers and is the Founding Editor of the Journal of Risk and Reliability, which now forms part O of the IMechE Proceedings.