# An algorithm for automated cause-consequence diagram construction

A. Valaityte & S.J. Dunnett
*Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough, Leicestershire, UK*

ABSTRACT: The cause-consequence diagram (CCD) method is a system safety technique which determines the logical combinations of causes of events, by the use of fault trees, and identifies all possible consequences of the events. Traditionally cause-consequence analysis is based on the manual construction of the CCD which is very time consuming, expensive and also a source of human errors. A way of overcoming these drawbacks is to have an automated method of constructing the diagram. Hence in this paper the development of an automated CCD construction algorithm is presented. The algorithm created is based on methods previously developed for reliability techniques. Using a model for each component in the system a set of rules are developed which automatically construct the CCD in an efficient manner. The procedure has been validated by testing it on a variety of industrial systems.

## 1 INTRODUCTION

Cause-consequence diagrams (Nielsen 1975) were developed as a graphical tool for the analysis and description of relevant accidents in a complex nuclear power plant and have subsequently been used in many industrial systems. The diagrams consist mainly of decision boxes which contain component/subsystem conditions. Following the YES/NO branches of the decision boxes the CCD is developed until the branches terminate in consequence boxes. They are attractive to engineers as the diagram can be constructed directly from the system description and it contains a full textual description of the systems behaviour. The exact system reliability can be obtained from the diagram in a very efficient calculation procedure. However, for many situations the diagram obtained will be very large and its final form is dependent upon the analyst constructing it. A faster and error free analysis could be performed if the CCD could be automatically generated by computer from the system description (Nielsen & Runge 1974).

Techniques have been developed previously which automate the construction and analysis of fault trees, see Lapps & Powers (1977), Salem et al. (1976), Xie et al. (1993), Henry & Andrews (1997), etc. These techniques have been extended in the work presented here to automate the construction of the CCD.

## 2 AUTOMATED CCD CONSTRUCTION & ANALYSIS ALGORITHM

The aim of the algorithm described in this work is to automatically construct a complete CCD for any industrial system which accurately models the physical behaviour of the system itself. The automatic generation is based on the following steps:

### 2.1 *Requirements*

In order to generate a CCD, three basic types of information are required: the component models, system topology diagram and the failure rate data. The component models are in the form of decision tables (Salem et al. 1976), which give a description of how the input and internal operational modes of each component influence the output states of that component. The system topology diagram describes how the various components of the system are interconnected. The failure rate data includes a description of the failure rates and failure modes of the components, and is used in the final CCD quantification. For some systems, initial states for the components are also specified.

In order to begin the construction process an initiating event should be identified and given by the user. It is generally changes to the state of a component which initiates the system. In the algorithm presented here that is represented by a component with an associated function. In order to ensure that the CCD construction process completes and that the algorithm reaches a consequence, stopping criteria must also be identified.

### 2.2 *Construction algorithm*

The CCD is constructed by applying a set of rules which have been developed in the course of this study. These rules have not been listed here in order to present the work in a concise manner, however some of them are included and applied in the example in section 3.3. In summary, initially the order in which components are considered is determined by use of the topology diagram. If the system contains circuits, these are considered first. A circuit is a path starting and ending at the same component which contains a power supply, and with all components passing current. Once the order is determined the CCD is constructed starting with the initiating event. The functionality of each component or subsystem, taken in the order determined, is then investigated using the decision tables and topology diagram. The procedure continues until the stopping criteria is reached, i.e. the consequences of investigated sequences determined.

### 2.3 *Reduction*

The algorithm then reduces the CCD to its minimal form. Each decision box in CCD is inspected and if any is deemed irrelevant (e.g. the branches attached to the NO and YES branches are identical) then this box is removed and the next decision box or consequence box in the path put in its place. When no further redundancies exist the cause-consequence diagram is deemed minimal (Andrews & Ridley 2001).

### 2.4 *Development of fault trees and analysis*

If a decision box in the CCD is governed not by a component but by a sub-system then the failure probability will be obtained via a fault tree (Andrews & Ridley 2002). The fault trees are produced automatically using fault tree construction methods developed previously (Salem et al. 1976, Henry & Andrews 1997).

Having constructed the CCD it can be used to analyse the system. A qualitative analysis produces the list of causes for each outcome. These are established by considering each decision box on a path to the particular outcome condition and listing the states of the components as indicated by the exit path from the decision box.

A quantitative analysis produces each system outcome probability. These are obtained by simply multiplying the probabilities of the component events in the branch leading to that consequence, since the algorithm ensures that the probabilities of the decision boxes of the CCD are independent (Nielsen 1975).

## 3 APPLICATION – FLUE DAMPER SYSTEM

To demonstrate the proposed algorithm it is applied to a flue damper system.

### 3.1 *System description*

Gas-fired storage water heaters are equipped with a device hood that regulates the flow or circulation of air connecting the flue to the chimney. Electric flue dampers open before the burner starts to allow combustion products to vent up the flue and close when the water is up to temperature to stop heat escaping. In this example, the operation “to fire burner” is considered. As shown in Figure 1, the flue damper electrical system consists of a relay and its contacts, the damper motor, a thermostat, an end switch, the burner control unit and the power supply. The initial states of the components are: damper closed, relay contacts close, thermostat open, end switch open. The system then operates by closing the thermostat due to the low temperature being reached. The relay is then energised and its contacts open stopping the motor. The spring drives the damper blade open which closes the end switch. Since the current reaches the burner control unit the burner fires.

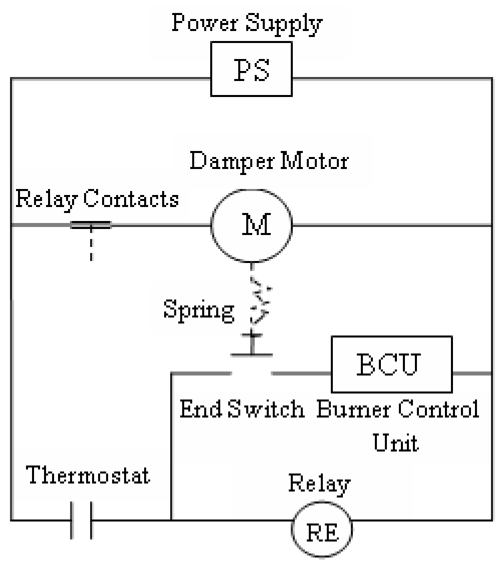


Figure 1. The flue damper electrical system.

Following the automated CCD construction algorithm outlined above the following steps are taken.

### 3.2 *Algorithm inputs*

A topology diagram for the flue damper system is constructed as shown in Figure 2.

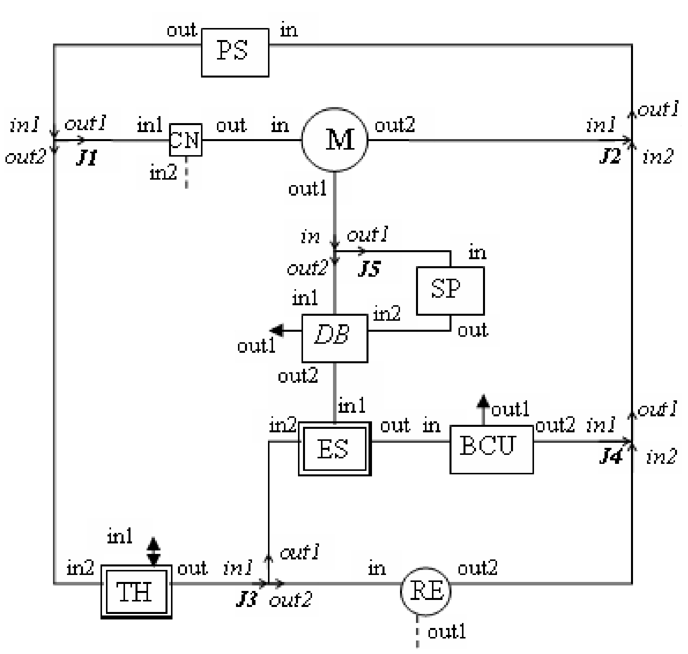


Figure 2. A topology diagram for the flue damper system.

Where TH is the thermostat, RE is the relay, PS is the power supply, M is the motor, SP is the spring, DB is the damper blade, ES is the switch, BCU is the burner control unit, CN are the relay contacts and J1-J5 are junctions 1-5 respectively. In Figure 2 OUT1 of RE connects to IN2 of CN.

There are eleven decision tables relevant to the components in the example, see Tables 1-11 below. As an example of how to construct a decision table for a component, consider the thermostat, see Table 1, which closes or opens according to the temperature. Two failure modes have been considered here, failed open (FO), when the component fails so that no connection is made, and failed closed (FC), when it cannot open. The number of failure modes considered is determined by the analyst. From the topology diagram, Figure 2, it is seen that TH has two inputs, IN1, – the automatic closing or opening according to the temperature and, IN2, the current from PS. IN1 has two possible states, closed (CL) and open (OP), and IN2 has current (C) or no current (NC). Table 1 considers all possible combinations of inputs from IN1 and IN2 and all possible states of TH and the effects these will have on the output to the relay or end switch. The sign “–” in the inputs and state columns indicates the “don’t matter” condition meaning that the specified input states will result in the specified output state regardless of the value of the variable. Considering the two causes of current in the output, from Table 1 it can be seen that 2 rows give this result:
row 1: input 1 is closed (IN1 = CL), current in input 2 (IN2 = C) and thermostat working (W);
row 5: current in input 2 (IN2 = C) and the thermostat failed closed (FC).

The decision table for the thermostat has two functions, which change the input IN1 when the *if* condition is satisfied, these are listed below the table. These two functions state that:

1. If the thermostat is working and the temperature is high (HT) then the component opens (IN1 => OP);
2. If the thermostat is working and the temperature is low (LT) then the thermostat closes (IN1 => CL).

The decision tables for the other components have been constructed by using the same technique.

Table 1. Decision table for the thermostat.

| | IN1 | IN2 | State | OUT |
|---|---|---|---|---|
| 1 | CL | C | W | C |
| 2 | OP | – | W | NC |
| 3 | – | NC | – | NC |
| 4 | – | – | FO | NC |
| 5 | – | C | FC | C |

(1) TH: If (State = W ∩ Temp = HT), then IN1 => OP;
(2) TH: If (State = W ∩ Temp = LT), then IN1 => CL.

Table 2. Decision table for the end switch.

| | IN1 | IN2 | State | OUT |
|---|---|---|---|---|
| 1 | CL | C | W | C |
| 2 | OP | – | W | NC |
| 3 | – | NC | – | NC |
| 4 | – | – | FO | NC |
| 5 | – | C | FC | C |

Table 3. Decision table for the burner control unit.

| | IN | State | OUT1 | OUT2 |
|---|---|---|---|---|
| 1 | C | W | ON | C |
| 2 | C | F | OFF | C |
| 3 | NC | – | OFF | NC |

Table 4. Decision table for the power supply.

| | IN | State | OUT |
|---|---|---|---|
| 1 | C | W | C |
| 2 | – | F | NC |
| 3 | NC | – | NC |

Table 5. Decision table for the junctions 1-4.

| | IN1 | IN2 | OUT1 | OUT2 |
|---|---|---|---|---|
| 1 | C | – | C | |
| 2 | – | C | C | |
| 3 | NC | NC | NC | |
| 4 | C | | C | C |
| 5 | NC | | NC | NC |

Table 6. Decision table for the relay.

| | IN | State | OUT1 | OUT2 |
|---|---|---|---|---|
| 1 | C | W | OP | C |
| 2 | NC | W | CL | NC |
| 3 | – | FO | OP | – |
| 4 | – | FC | CL | – |
| 5 | C | – | – | C |
| 6 | NC | – | – | NC |

Table 7. Decision table for the contacts.

| | IN1 | IN2 | OUT |
|---|---|---|---|
| 1 | C | CL | C |
| 2 | – | OP | NC |
| 3 | NC | – | NC |

Table 8. Decision table for the motor.

| | IN | State | OUT1 | OUT2 |
|---|---|---|---|---|
| 1 | C | W | ON | C |
| 2 | C | F | OFF | C |
| 3 | NC | – | OFF | NC |

Table 9. Decision table for the junction 5.

| | IN | OUT1 | OUT2 |
|---|---|---|---|
| 1 | ON | ON | ON |
| 2 | OFF | OFF | OFF |

Table 10. Decision table for the spring.

| | IN | State | OUT |
|---|---|---|---|
| 1 | OFF | W | OP |
| 2 | OFF | F | CL |
| 3 | ON | – | CL |

Table 11. Decision table for the damper blade.

| | IN1 | IN2 | OUT1 | OUT2 |
|---|---|---|---|---|
| 1 | ON | – | CL | OP |
| 2 | – | OP | OP | CL |
| 3 | OFF | CL | CL | OP |

Where ON and OFF denotes motor turned on and off respectively, and F is failed. The blank cells in the decision table for the junctions 1-4 are irrelevant.

The failure data for the components in the flue damper system is shown in Table 12. Failure rates have not been included as the example has been taken in order to demonstrate the construction process.

Table 12. Failure data for the components.

| Component | Failure mode | Description |
|---|---|---|
| TH | TH_FO | Thermostat failed opened, it doesn't close when temperature is low |
| | TH_FC | Thermostat failed closed, it doesn't open when temperature is high |
| ES | ES_FO | End switch failed opened, damper blade cannot close it |
| | ES_FC | End switch failed closed, it cannot be opened |
| BCU | BCU_F | Burner control unit failed, it doesn't fire |
| PS | PS_F | Power supply failed, no current to the circuits |
| RE | RE_FO | Relay contacts fails opened |
| | RE_FC | Relay contacts fails closed |
| M | M_F | Motor failed, damper blade cannot be closed |
| SP | SP_F | Spring failed, damper blade cannot be opened |

The initial conditions, are that the thermostat and the end switch are open, and the relay contacts and the damper blade are closed, i.e.:
TH: IN1 = OP;
ES: IN1 = OP;
RE: OUT1 = CL;
DB: OUT1 = CL.

The initiating component and its function must now be identified. The system considered is initiated when the thermostat closes. Therefore the initiating event is taken to be the thermostat TH with the temperature assumed to be low, i.e.:
TH: Temp = LT.

The stopping criteria for the operation "to fire burner" is that all the components in the system which have the capability to change their outputs have been considered. Hence every path of the diagram terminates in one of the following consequences:

- normal start, i.e. system works normally;
- failed to start, i.e. damper blade opens, but burner doesn't fire;
- no start, i.e. damper blade remains closed and burner doesn't fire.

Thus the consequence box will be governed by the output of the damper blade and the burner control unit. If neither of these components are considered while constructing any of the CCD branches, the consequence box will show their initial outputs.

The algorithm now has all the information it requires to generate the CCD.

## 3.3 *Construction of CCD*

As mentioned in section 2.2 the CCD is constructed by applying a set of rules. Some of these rules relevant to the example are listed below to demonstrate the procedure:

1. Initially the algorithm decides the order in which the components are to be considered. Starting from the initiating event, components are traced through the topology diagram and if any circuits are identified these are numbered. If a component is within a circuit and its output is connected to another component which is contained within a different circuit, then, these two circuits must be considered in turn when constructing the CCD.
2. If the considered component and its function are traced in a circuit the decision box is created asking about current or no current in the circuit according to expected condition of considered component, else the decision box added to the CCD is related to the output of the component obtained from the components decision table.

3. The output of the component which is considered is obtained by taking the rows in the decision table in turn with the “don’t matter” cells ignored and considered last with such cells in inputs or state of the component.
4. If a circuit contains any components with external output then these should be considered first.
5. When the output of a component is set, from the topology diagram the component connected to this output is traced and hence an input in the decision table for this component set. Then all remaining inputs and the state of this following component are considered in the order in which the columns and then the rows follow in the table.
6. If there are “don’t matter” cells in the inputs or state of the row of the decision table considered, these should be checked for consistency with assumptions made so far.
7. If the given inputs for the component considered in a row of the decision table result in outputs which match previous ones then consider the next row of the table. If there are no more rows consider the component connected to the considered components output.
8. If a component is repeated in the same branch of the CCD with the same inputs and outputs, then a consequence box is created.
9. When the stopping criteria is reached – create a consequence box indicating the result.
10. The CCD is constructed by completing the YES branches of the decision boxes first until a consequence box is reached and then back tracking to the last decision box and developing the NO branch in the same way.

In the example applying rule 1 it can be seen that there are three circuits: {PS, J1, CN, M, J2, PS}, {PS, J1, TH, J3, ES, BCU, J4, J2, PS} and {PS, J1, TH, J3, RE, J4, J2, PS}. As the circuits contain different components they are treated individually.

Since the thermostat TH is the initiating component and the input is Temp = LT, the conditions for the second function of the thermostat are satisfied only if TH is in a working state, see Table 1.

Assuming that TH is working so that IN1 => CL, the algorithm searches for the circuits which contain this component, these are: 2nd {PS, J1, TH, J3, ES, BCU, J4, J2, PS} and 3rd {PS, J1, TH, J3, RE, J4, J2, PS}. Hence these are investigated in turn starting with the 2nd circuit. The algorithm searches for the rows in Table 1 whose first column for the input IN1 could take the value CL. These are rows 1, 3, 4 and 5. Considering row 1, IN2 = C. The algorithm then traces the other components in circuit 2. TH is connected to ES and for this component the initial conditions state that IN1 = OP. Hence only rows 2-4 in Table 2 are satisfied. These all result in OUT = NC and hence there is no current in the 2nd circuit. No decision box relating to circuit 2 is therefore necessary. The topology diagram (Fig. 2) is then checked to determine if any components within circuit 2 have outputs that are external and therefore connect to components within other circuits or parts of the system, see rule 4. Only one component is identified, the burner control unit with external output OUT1. As IN of BCU is NC row 3 is satisfied resulting in OUT1 = OFF, see Table 3.

Circuit 3 is investigated next and rule 2 applied. As row 1 in Table 1 for TH is being considered, decision box 1 is created related to the output OUT = C, see Figure 3a: “Is C in Circuit 3?” i.e. does the thermostat close and power supply work?

The YES branches of the decision boxes are traced first until the consequence box is reached, according to the rule 10 developed. Therefore the YES branch of decision box 1 is traced which results in current in the 3rd circuit, hence there is current C in OUT of PS (row 1 in Table 4), OUT2 of J1 (row 4 in Table 5), OUT of TH (row 1 in Table 1), OUT2 of J3 (row 4 in Table 5), OUT2 of RE (rows 1, 3-5 in Table 6), OUT1 of J4 (row 2 in Table 5) and OUT1 of J2 (row 2 in Table 5). Any components within the 3rd circuit with external output are then considered (rule 4). There is only one, RE, with external output OUT1. Since C is in IN, four matches are found (rows 1, 3, 4 and 5, Table 6). However, in the 3rd row the relay is in a failed opened state, which contradicts the initial assumption that the relay is closed. Also conditions in row 5 do not affect OUT1 and hence are not considered at this point as stated in rule 3. Considering the remaining rows in turn row 1 is considered first and in this case OUT1 = OP, hence the decision box contains the question “Is RE: OUT1 = OP?” (rule 2).

According to rule 5, the algorithm proceeds by tracing the component connected to the external output of relay. From the topology diagram, Figure 2, it can be seen that output OUT1 of RE connects to input IN3 of contacts CN, which is within circuit 1: {PS, J1, CN, M, J2, PS}. By the conditions already considered, i.e. the power supply being in a working state and the relay contacts being opened (row 2 in Table 7), following the YES branch of decision box 2, circuit 1 doesn’t pass current. The external output OUT1 of motor M is considered next (rule 4), but no decision box is created since there is only one possible outcome in this case (rule 7), see row 3 in Table 8.

Since all possible circuits and components with external outputs have been investigated, the functions of TH can now be checked, but none of them are implemented as the temperature has not changed.

Following the YES branch of decision box 2, the algorithm proceeds by considering the output of the spring, as OUT1 of M is connected to IN of SP via junction J5 (row 2 in Table 9). Two rows, 1 and 2, of the decision table for the spring (Table 10) are applicable as IN = OFF. Considering row 1 decision box 3 is created “Is SP: OUT = OP?” (rule 2), see Figure 3b. By examining the system topology diagram (Fig. 2) it can be seen that OUT1 of M and OUT of SP are connected to the inputs, IN1 and IN2, of DB, respectively. Following the YES branch of box 3 results in the 2nd row of the decision table for the damper blade being the appropriate row, see Table 11.

As OUT2 of damper blade is connected to IN1 of end switch the component ES is considered next, which is within circuit 2. Four rows (rows 1, 3-5) of the decision table for ES (Table 2) are applicable as IN1 = CL. Since the initial condition for the end switch was IN1 = OP the component cannot have failed closed and hence the 5th row is not possible (rule 6). Considering row 1 in Table 2 decision box 4: “Is C in Circuit 2?” is created (rule 2), see Figure 3b. Following the YES branch component BCU is considered next as according to rule 4 it has external output OUT1.  Rows 1 and 2 in Table 3 satisfy the criteria, and row 1 is considered first. Hence the decision box 5 is created with the question related to the output OUT1 in row 1: “Is BCU: OUT1 = ON?” (rule 2), see Figure 3b.

The stopping criteria is now reached as all the necessary components in the system have been checked. Hence, following rule 9, the YES branch of the CCD terminates in a consequence box indicating a *normal start* as damper blade is open (DB: OUT1 = OP) and burner control unit is on (BCU: OUT1 = ON), see Figure 3b.

The algorithm proceeds by returning to the NO outputs of the last decision box (rule 10), which in this case is box 5. Following the NO branch all possible values of OUT1 in Table 3, except for ON, must be considered with IN = C from box 4. There is only one other value, OFF, and only row 2 coincides with the given situation. As there are no other options for the components considered the stopping criteria is reached and according to rule 9 a consequence box “failed to start” is created (i.e. DB: OUT1 = OP, BCU: OUT1 = OFF), see Figure 3b.

The rest of the diagram is obtained in the same manner by back tracking to the last decision box and developing the NO branch until the diagram is complete (rule 10).

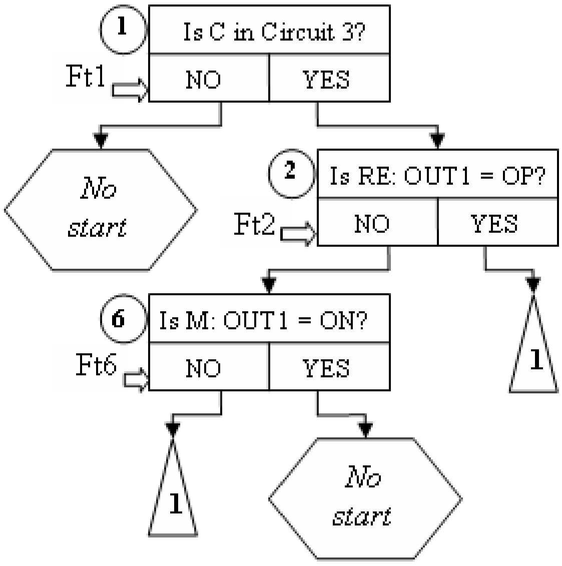


Figure 3a. Cause-consequence diagram for the flue damper system.

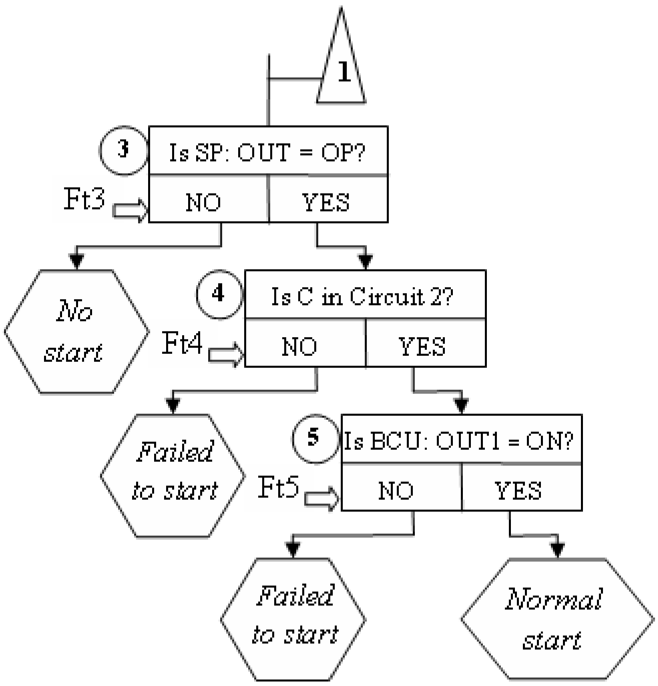


Figure 3b. Cause-consequence diagram for the flue damper system.

### 3.4 *Reduction of CCD*

The cause-consequence diagram presented in Figures 3a, b cannot be reduced as no redundant decision boxes are identified.

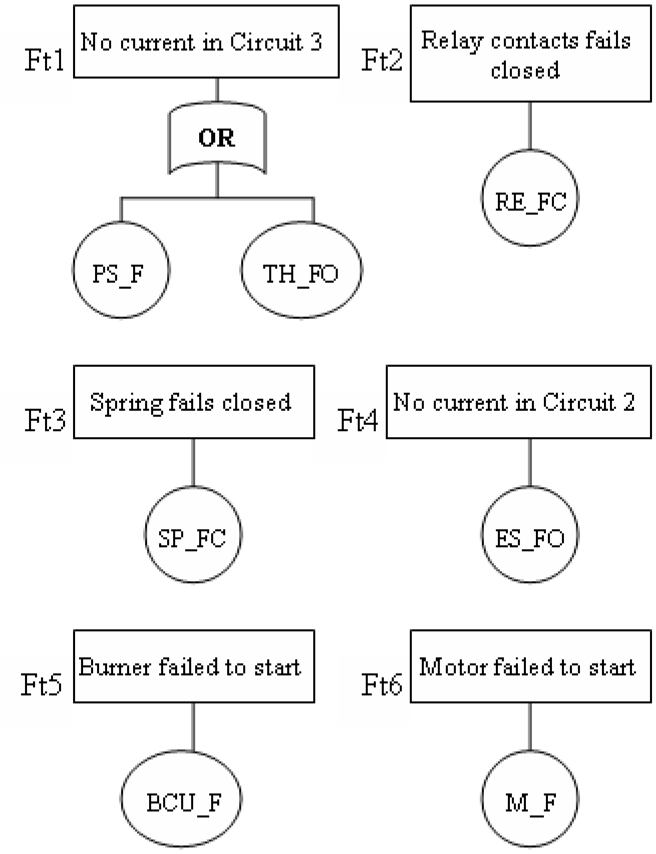


Figure 4. Fault trees for the CCD shown in Figures 3a, b.

### 3.5 *Development of fault trees*

Fault trees are now developed according to the decision table method for each decision box starting with the first one and these are shown in Figure 4.

### 3.6 *Qualitative and quantitative analysis*

Having obtained the diagram it can now be analysed and quantified in a straightforward manner.

For the flue damper system considered the failure event is "failed to start", i.e. the consequence DB: OUT1 = OP, BCU: OUT1 = OFF. There are four consequence boxes with this outcome in the diagram. The failure events leading to these boxes must be traced to obtain the minimal cut sets, e.g. for the consequence box on the NO branch of decision box 5 the component failures leading to this are given from Ft5 (fault tree 5), i.e. BCU_F. Considering the other consequence boxes in the same manner leads to the complete list of minimal cut sets: {BCU_F}, {ES_FO}, while {RE_FC, M_F, BCU_F} and {RE_FC, M_F, ES_FO} are non minimal. The probability of the outcome "failed to start" can be obtained by adding the probability of each path:

$$
\begin{aligned}
&\mathrm{P(Failed\ to\ start)} = \mathrm{P(Path\ 1)} + \mathrm{P(Path\ 2)} + \\
&+ \mathrm{P(Path\ 3)} + \mathrm{P(Path\ 4)} = \\
&= (1-q_{Ft1})(1-q_{Ft2})(1-q_{Ft3})(1-q_{Ft4})q_{Ft5} + \\
&+ (1-q_{Ft1})(1-q_{Ft2})(1-q_{Ft3})q_{Ft4} + \\
&+ (1-q_{Ft1})q_{Ft2}q_{Ft6}(1-q_{Ft3})(1-q_{Ft4})q_{Ft5} + \\
&+ (1-q_{Ft1})q_{Ft2}q_{Ft6}(1-q_{Ft3})q_{Ft4} = \\
&= (1-q_{TH\_FO}-q_{PS\_F}+q_{TH\_FO}q_{PS\_F})\cdot \\
&\cdot (1-q_{RE\_FC})(1-q_{SP\_FC})(1-q_{ES\_FO})q_{BCU\_F} + \\
&+ (1-q_{TH\_FO}-q_{PS\_F}+q_{TH\_FO}q_{PS\_F})\cdot \\
&\cdot (1-q_{RE\_FC})(1-q_{SP\_FC})q_{ES\_FO} + \\
&+ (1-q_{TH\_FO}-q_{PS\_F}+q_{TH\_FO}q_{PS\_F})\cdot \\
&\cdot q_{RE\_FC}q_{M\_F}(1-q_{SP\_FC})(1-q_{ES\_FO})q_{BCU\_F} + \\
&+ (1-q_{TH\_FO}-q_{PS\_F}+q_{TH\_FO}q_{PS\_F})\cdot \\
&\cdot q_{RE\_FC}q_{M\_F}(1-q_{SP\_FC})q_{ES\_FO}
\end{aligned}
$$

where $q_{TH\_FO}$ is the probability of thermostat (TH) failure to close (fails open); $q_{PS\_F}$ is the probability of power supply (PS) failure; $q_{RE\_FC}$ is the probability of relay (RE) failure to open the contacts (fails closed); $q_{SP\_FC}$ is the probability of spring (SP) failure to open the damper blade; $q_{ES\_FO}$ is the probability of end switch (ES) failure to close; $q_{BCU\_F}$ is the probability of burner control unit (BCU) failure and $q_{M\_F}$ is the probability of motor (M) failure.

Although this is only a small example it has features typical of a larger industrial system.

## 4 CONCLUSIONS

An automated cause-consequence diagram construction algorithm has been proposed in this paper. Employing the system topology diagram and decision tables as well as methods developed previously for fault tree construction the algorithm develops the CCD in an efficient manner. There is a set of automatic CCD construction rules given here, but for reading conciseness they have not been included in the text. Once the diagram is obtained the probabilities of all the possible outcomes can be easily quantified.

In this paper the algorithm is applied to a simple example in order to demonstrate the feasibility of the approach. It has also been applied to several substantial industrial examples and found to perform well.

## REFERENCES


Andrews, J.D. & Moss, T.R. 2002. *Reliability and Risk Assessment, 2nd ed*. London and Bury St. Edmunds: Professional Engineering Publishing.

Andrews, J.D. & Ridley, L.M. 2001. Reliability of sequential systems using the cause-consequence diagram method. *IMechE Proceedings Part E: Journal of Process Mechanical Engineering* 215(3): 207-220.

Andrews, J.D. & Ridley, L.M. 2002. Application of the cause-consequence diagram method to static systems. *Reliability Engineering and System Safety* 75(1): 47-58.

Henry, J.J. & Andrews, J.D. 1997. A computerised fault tree construction methodology. *IMechE Proceedings Part E, Journal of Process Mechanical Engineering* 211(3): 171-185.

Lapp, S.S. & Powers, G.J. 1977. Computer-aided synthesis of fault trees. *IEEE Transactions on Reliability* 26(1): 2-13.

Nielsen, D.S. 1975. Use of cause-consequence charts in practical systems analysis. *Reliability and Fault Tree Analysis*: 849-880.

Nielsen, D.S. & Runge, B.A. 1974. Unreliability of a standby system with repair and imperfect switching. *IEEE Transactions on Reliability* 23(1): 17-24.

Salem, S.L., Apostolakis G.E. & Okrent D. 1976. A computer-oriented approach to fault tree construction. *EPRI Report*: 288.

Xie, G, Xue, D. & Xi, S. 1993. Tree-Expert: a tree based expert system for fault tree construction. *Reliability Engineering and System Safety* 40(3): 295-309.