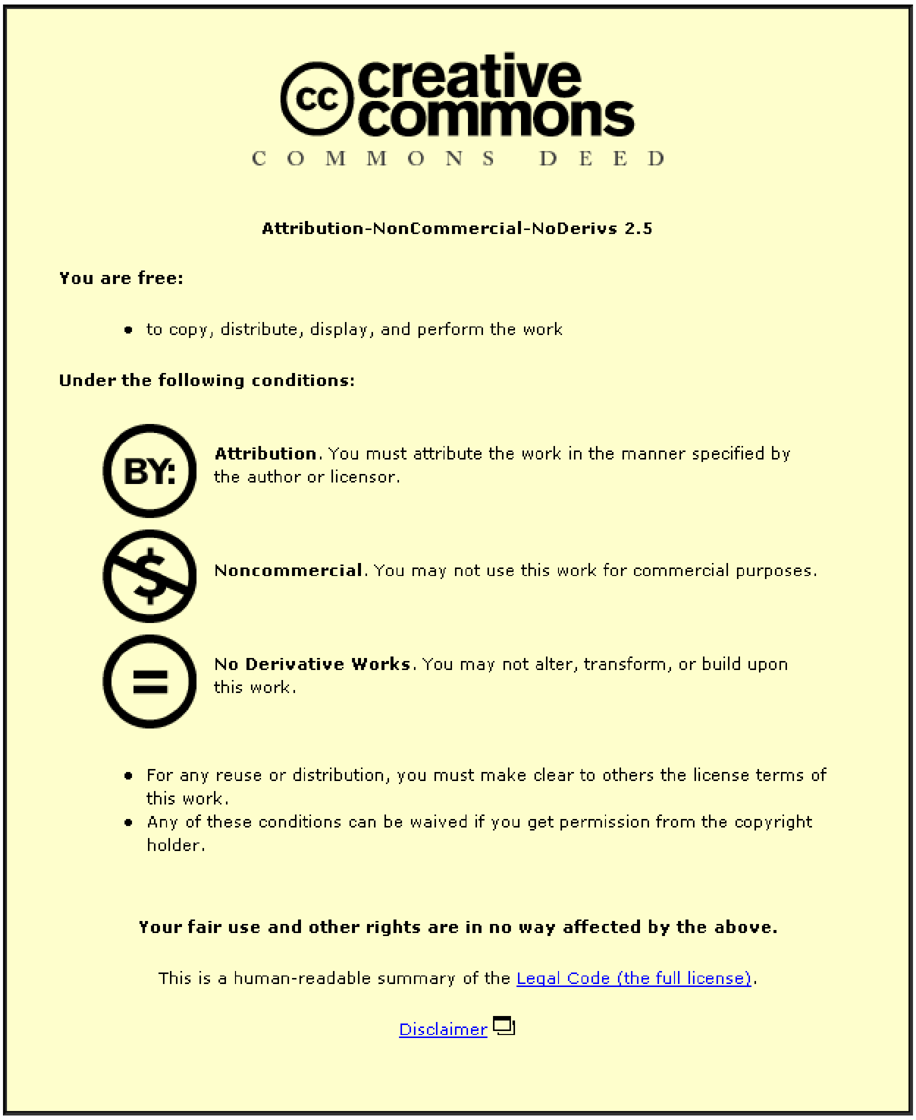


A Branching Search Approach to Safety System Design Optimisation

Prof. J.D.Andrews & Dr. L.M.Bartlett*;Department of Aeronautical and Automotive Engineering; Loughborough University; Loughborough;Leicestershire; LE11 3TU; UK

Abstract

Safety systems are designed to prevent or mitigate the consequences of potentially hazardous events. In many industries the failure of such systems can result in fatalities. Current design practice is usually to produce a safety system which meets a target level of performance that is deemed acceptable by the regulators. However, when the system failure will result in fatalities it is desirable for the system to achieve an optimal rather than adequate level of performance given the limitations placed on available resources.

The unavailability of safety systems can be predicted using fault tree analysis methods. Formulating an optimisation problem for the system design has features which make standard mathematical optimisation techniques inappropriate. The form of the objective function is itself a function of the design variables, the design variables are mainly integers and the constraint forms can be implicit or non-linear.

This paper presents a Branching Search algorithm which exploits characteristics common to many safety systems to explore the potential design space and deliver an optimal design. Efficiency in the method is maintained by performing the system unavailability evaluations using the Binary Decision Diagram method of fault tree solution. Limitations are placed on resources such as cost, maintenance down-time and spurious trip frequency. Its application is demonstrated on a High Integrity Protection System.

Keywords: Optimisation, Safety Systems, Fault Tree Analysis

1. Introduction

The traditional design process consists of the iterative stages of preliminary design, analysis, appraisal and redesign. For safety systems the design is usually required to deliver a specified availability performance to keep the risk within tolerable limits. If, following analysis, the initial design does not meet the pre-determined acceptability criteria for system unavailability, it is redesigned and the analysis and appraisal repeated. Since safety system failure can, in many industries, result in fatalities a better approach would be to maximise the availability of the system and in turn minimise the potential fatalities. This would require an optimisation process to be used to determine the design parameters. The system performance is optimised subject to constraints placed on the resources available (such as initial cost, life-cycle cost, weight, volume, spurious trip frequency or maintenance effort). The variables concerned with the design and operation of safety systems will be features such as redundancy levels, diversity levels, component selection, voting systems for sensors and maintenance test intervals. The system assessment can be carried out using the fault tree analysis method embedded within the optimisation methodology.

The safety system design optimisation problem has features that mean generalised mathematical optimisation procedures are either inappropriate or ineffective. In the main difficulties arise due to the lack of an accurate explicit objective function over the entire, or even relatively small part, of the search space. This is caused because each time a design parameter is altered a new objective function is formed. Further complications occur as a result of the integer stipulation assigned to all of the design variables, where the integer values are usually low and their range restricted. The constraints limiting the available resources can also be highly non-linear in form and in some instances, for example the spurious trip frequency, are implicit and require a full system evaluation to assess. In addition, certain design parameters are dependent on the values assigned to other parameters and an optimisation method which manipulates these variables independently, may result in an infeasible optimal design.

An approach by which optimal performance can be obtained using the Fault Tree Analysis method to determine the availability of each system was first introduced in 1994 [1]. A single fault tree was developed to represent the causes of system failure for all possible design variations. This was achieved using House Events to turn on and off appropriate branches. This approach has since been modified and improved by using the Binary Decision Diagram method [2-6], to analyse the fault trees. This makes the analysis procedure more efficient and more accurate. The optimisation procedures which can be used to perform the optimisation include Genetic Algorithms [7-8], the Grid-Sampling technique [9] and a design of experiments based method [10]. An efficient optimisation procedure would need to be problem specific and developed to solve each particular design problem. Some integer programming problems can be solved by Branch and Bound algorithms and it is such an approach that is presented in this paper.

The remainder of this paper discusses reasons for using Branching Search (section 2), the application to a safety system design problem (section 3), safety system design performance (section 4), the Branching Search Algorithm (section 5), and the results of applying this new technique to the safety system are discussed in the final sections.

2. Design Optimisation Procedure

Certain optimisation techniques address one or more of the difficult features of the safety system design problem such as: integer variables, no explicit objective function, implicit constraints and the interaction of parameters. A specialised method, or methods, could, thus, be created to solve the safety system design optimisation problem using an amalgamation of features from the applicable techniques. Aspects typical of safety system design such as redundant/diverse structures give some independent partitioning of the design space which can be used to focus the method developed to the specific problem. The key features of different optimisation techniques that lend themselves to the safety system optimisation are discussed below.

When little is known about the function in question the aim is to devise an optimisation algorithm that will distinguish between the local minima and locate the best local minimum, i.e. solve the global optimisation problem. In the majority of cases this is successfully achieved using probabilistic methods. The simplest of these is the random search. The use of random sampling of decision variables within their specified range is applied in much of the random search literature. This feature is a simple yet effective means of establishing a feasible solution vector, being easy to implement within a general optimisation algorithm. In addition random sampling encourages global consideration of the search space thus combating premature convergence of a localised approach.

Optimising integer values within a restricted range has been achieved by using a 'Combinatorial heuristic' [11]. The method relies on fixing design variables using a look-ahead search, which is a means to incorporate interaction with the next variable. The method uses little or no assumption concerning the structure of the objective function and constraint functions and constraint evaluations can easily be incorporated at each step with infeasible designs rejected. Of most importance is that the approach is able to consider an element of dependency between the system variables.

Misra and Sharma in 1991 [12] proposed a simple and effective technique for solving integer programming problems. It involved a systematic search near the boundary of the constraints. 'Tolerable slacks' were introduced to define the appropriate regions to search given the constraints. For techniques which are very computationally intensive these tolerable slacks limit the number of function evaluations required as the optimal design will usually fully utilise all available resources and the interior of the design space can be ignored. This will also enable designs for which there is a significant improvement in performance for a minor violation of a constraint to remain in contention for the chosen design.

For the system design optimisation problem, the use of random sampling should help to find the global solution to the problem. Incorporating the combinatorial heuristic look-ahead search type approach allows for integer values and variable dependency, and if tolerable slacks are included, this restricts the number of function evaluations required. The combination of these three elements should provide a useful tool for this specific optimisation problem, and will be referred to as the Branching Search approach.

3. Branching Search Approach and Application Safety System

The Branching Search approach uses random sampling and a look-ahead search combined with the use of tolerable slacks about the constraints to improve the efficiency of the search. This approach is to be tested on a real industrial problem involving a High Integrity Protection System (HIPS). The function of the HIPS is to prevent a high-pressure surge passing through the system. The structure of the system is shown in figure 1. The high pressure originates from a production well of a not normally manned offshore platform (left of the diagram) and the pieces of equipment to be protected are vessels located downstream on the processing platform (right of the diagram).

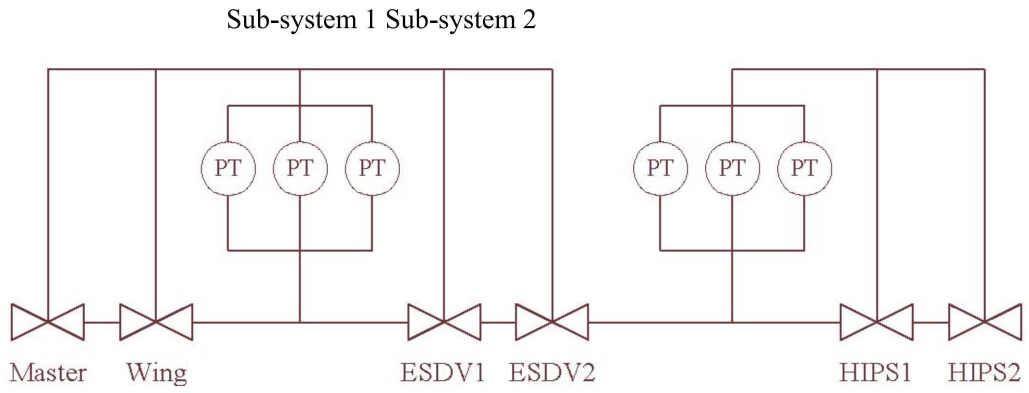


Figure 1 -High Integrity Protection System

The system has two levels of protection which function independently. The first is the emergency shutdown (ESD) subsystem and the second is a high-integrity protection system (HIPS). Both work in a similar manner. Pressure in the pipeline is monitored using pressure transmitters (PT's). When the pipeline pressure exceeds the permitted value then the ESD system acts to close the Wing and Master valves on the well together with any ESD valves that have been fitted. If the pressure transmitters in the HIPS system detect an excess in pressure this system acts to close the HIPS valves.

Within this system the problem is to determine the necessary configuration of the high integrity protection. As with many practical applications there is a conflict between operational and safety requirements. The aim of this method is therefore to optimise the system design parameters to minimise system unavailability, given limited resources.

To minimise the system unavailability a number of design options have been considered. These are:

i) How many ESD valves are required (0,1,2)? E

ii) How many HIPS valves are required (0,1,2)? H

iii) How many pressure transmitters for each subsystem (0,1,2,3,4)? N1, N2

iv) How many transmitters are required to trip? K1, K2

v) Which of two possible types of ESD/HIPS valves to select? V

vi) Which of two possible types of pressure transmitters to select? P

vii) Maintenance test interval in weeks for each subsystem (4-104 weeks)? θ_1, θ_2

Although a relatively small system, even with just these ten design parameters, it is not possible to manually trade off the effects of different design alternatives, hence the need for a technique which will provide the optimal solution.

The data relating to the system components are provided in table 1. This data specifies the failure rate (dormant and spurious failure modes), average repair time, cost, and test time.

A look-ahead search is carried out about each of the variables N1, N2, K1, K2, E and H. Interaction with the types of variables, V and P, and each maintenance test interval parameter θ_1 and θ_2, is considered within each look-ahead search.

In addition to the design variables, there are three constraints that have been placed on the design. These are limitations on: i) the total system cost, ii) the average time each year that the system resides in the down state due to preventive maintenance, and iii) the number of times that a spurious system shutdown occurs.

Component	Dormant Failure Rate	Dormant Mean Repair Time	Spurious Failure Rate	Spurious Mean Repair Time	Cost	Test time
Wing Valve	1.14 x 10^{-5}	36.0	1 x 10^{-6}	36.0	100	12

Master Valve	1.14 x 10_{-5}	36.0	1 x 10_{-6}	36.0	100	12
HIPS1	5.44 x 10_{-6}	36.0	5 x 10_{-7}	36.0	250	15
HIPS2	1 x 10_{-5}	36.0	1 x 10_{-5}	36.0	200	10
ESDV1	5.44 x 10_{-6}	36.0	5 x 10_{-7}	36.0	250	15
ESDV2	1 x 10_{-5}	36.0	1 x 10_{-5}	36.0	200	10
Solenoid Valve	5 x 10_{-6}	36.0	5 x 10_{-7}	36.0	20	5
Relay Contacts	0.23 x 10_{-6}	36.0	2 x 10_{-6}	36.0	1	2
PT1	1.5 x 10_{-6}	36.0	1.5 x 10_{-5}	36.0	20	1
PT2	7 x 10_{-6}	36.0	7 x 10_{-5}	36.0	10	2
Computer Logic	1 x 10_{-5}	36.0	1 x 10_{-5}	36.0	20	1

Table 1 -Component Data

Tolerable slacks must be established about each of the constraints. It is decided to set these bounds as follows:

800 units ≤ cost ≤ 1000 units

110 hours ≤ MDT ≤ 130 hours

0.6 per year ≤ spurious trip frequency ≤ 1 per year Designs are assumed infeasible when constraint values fall either side of the tolerable slacks.

4. Safety System Design Performance

The most important feature of a safety system is that it functions on demand. The objective is, therefore, to minimise system unavailability and as such this provides a measure of system performance. There is no explicit objective function that can be formulated to evaluate system unavailability. The fault tree method is utilised for this. House events [1] are used within the fault tree to allow for all potential designs to be incorporated within one tree. House events in the fault tree, which are either TRUE or FALSE, are utilised to switch on or off different branches to model the changes in the causes of failure for each design alternative. An example fault tree section for the causes of valve failure where the valve installed in the design can be one of two alternative types is shown in figure 2.

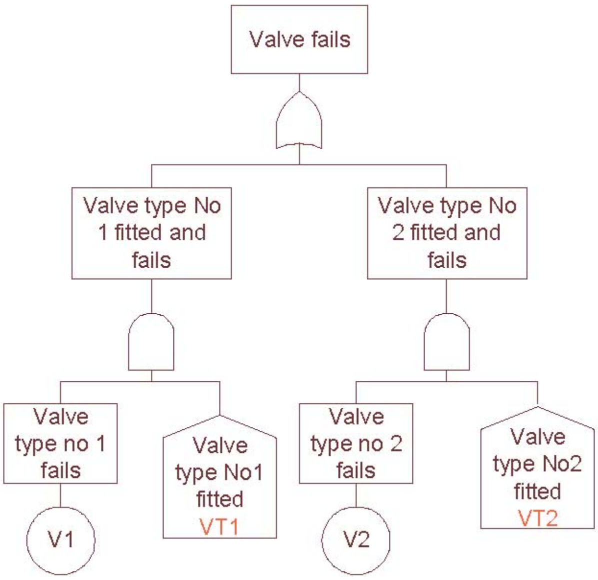


Figure 2 -Fault Tree with House Events to represent the design variables

Analysis of the fault tree structure is evaluated using the Binary Decision Diagram (BDD) Approach, which can introduce significant advantages into the quantitative process and allows the probability of the system failure mode to be calculated exactly rather than using approximate methods. Details of the fault tree to BDD conversion process and the analysis of the resulting logic diagram are given in reference [8].

5. Branching Search Algorithm

For any design problem a deep understanding of the system and its requirements is required prior to any detailed modelling. Thus, any optimisation technique implemented needs to be structured according to the problem. The Branching Search Algorithm is specific to the features of the HIPS safety system and is implemented in the following steps:

1) An initial design is randomly generated, and its unavailability is calculated using the BDD methodology. Each of the constraints are checked and where a violation occurs and the design is infeasible a new design is generated. This process is repeated until a feasible design is achieved.

2)A variable ordering is generated, i.e. the order in which the variables are considered in the look-ahead search. Initially the ordering is chosen to be {N1, K1, E, H, N2, K2, *E, K1, N1, H, K2, N2*}. The choice of ordering is somewhat arbitrary and as such offers a significant degree of leeway. Variables adjacent to one another are those that are most likely to interact, therefore aiding the look-ahead search. In the main an optimal design is established following a single iteration of each variable. An additional iteration of each variable is however, implemented and the variable order shuffled, as shown by the variables in italics in the initial ordering. This serves the purpose of considering further likely interactions. The variable ordering does make use of the independent redundant structures of the ESD and HIPS systems and places variables of each of the subsystems adjacent in the ordering.

3) For each variable in turn, as specified by the variable ordering, the look-ahead search progresses as given below: i) For the variable selected, a feasible range is established in view of the fixed values assigned to the other variables in the current design vector. ii) A look-ahead search is then conducted involving the next variable in the ordering list and either V or P. If the next variable in the list concerns the pressure transmitters (N1, N2, K1, K2) then P is chosen else (E, H) V is the variable to be considered. iii) All the possible combinations of the feasible values of the three variables are established. The remaining design parameters are held fixed with values as in the starting design. The associated system unavailability is generated as follows: a) Evaluate system cost, C. Check that the resulting value lies within the tolerable slacks for cost. If C < 800 or C > 1000 units assign a value of 1 to the system unavailability of the design in question and repeat part (a) with the next combination. If all combinations have been considered go to step (iv). b) Evaluate the spurious trip frequency, F. Ensure that the resulting value lies within the tolerable slacks for the trip rate. If F < 0.6 or F > 1 assign a

value of 1 to the

system unavailability of the design in question and repeat part (a) with the next combination. If all combinations have been considered go to step (iv).

c) For each combination of maintenance test intervals calculate the design's MDT. The maintenance test interval ranges from 4 – 104 weeks, in intervals of 4 weeks. This is used to limit the number of function evaluations needed. If the MDT falls within the respective tolerable slacks evaluate the system unavailability associated with the design. Retain the maintenance test interval combination that results in the most optimal system unavailability for the design in question. If MDT < 110 or MDT > 130 hours assign a value of 1 to the system unavailability of the design in question and repeat part (a) with the next combination. If all combinations have been considered go to step (iv).

iv)The system unavailability associated with each possible design is compared and the best is selected. If the current variable selected is the penultimate in the list, the algorithm continues at step (4). Else, the chosen variable is fixed, along with the type variable, and step (3) is repeated for the next variable in the ordering list.

4) The resulting best design, considers maintenance test interval values for subsystems 1 and 2 in 4 weekly intervals. Optimise the use of the available MDT resource for this resulting best design over the full range of test interval values.

5) Ensure termination criteria met. Else repeat step (3) with first variable in ordering list.

The order in which the variables are considered will affect the steps of the algorithm and may result in convergence to a different optimal design vector. The order of variables is considered in more detail in section 7.

There are various means to terminate the algorithm. Rather than specifying a fixed number of iterations the program could be altered to terminate when, for example, the optimal design has consistently attained a specified termination criteria over *k* iterations, where the ordering is repeated up to this point. The value of *k* is set to 3 for use in this research.

6. Application of Branching Search Algorithm to System Analysis

The Branching Search Algorithm was applied to the HIPS system. A detailed run through of the steps taken is given below with the output specified at each step.

Step 1) An initial design ($\mathbf{x}^0$) is randomly generated.

	E	K1/N1	H	K2/N2	P	V	θ1	θ2	Q_{SYS}	Q'_{SYS}
x(0)	1	2/3	1	2/2	1	1	23	36	0.00137	0.0061

Step 2) Variable ordering, {N1, K1, E, H, N2, K2, E, K1, N1, H, K2, N2} established.
Step 3) First variable in list considered, N1.

i) Feasible values for N1, given values of other parameters are {2, 3, 4}. N1 cannot equal 1 as the number of transmitters set to trip (K1) is 2.

ii) K1 is the next variable in the ordering list and as it relates to the pressure transmitters P is the type variable chosen.

iii) All possible combinations of N1, K1 and P are generated with E = 1, H = 1, K2/N2 = 2/2, V = 1 fixed as in the initial design. Each combination and its related fitness are shown in table 2. Note that in all but rows 5 and 11 the spurious trip frequency lies outside its respective tolerable slacks and hence, values are not assigned to the maintenance test interval parameters.

iv) The best design arises in the fifth row of table 2. This design is now used for consideration of the next variable in the ordering list, with N1 = 3 and P = 1.

Step (3) is repeated with the next variable in the ordering list.

x(0) N1,K1,P	**K1/N1**	**P**	**θ11**	**θ22**	**Cost**	**F_{SYS}**	**MDT**	**Q_{SYS}**
1	1 / 2	1	-	-	902	0.551	-	1

2	1 / 2	2	-	-	862	1.51	-	1
3	2/2	1	-	-	902	0.289	-	1
4	2/2	2	-	-	862	0.295	-	1
5	1/3	1	48	20	922	0.681	130	0.0016
6	1/3	2	-	-	872	2.11	-	1
7	2/3	1	-	-	922	0.289	-	1
8	2/3	2	-	-	872	0.301	-	1
9	3/3	1	-	-	922	0.290	-	1
10	3/3	2	-	-	872	0.292	-	1
11	1 / 4	1	52	20	942	0.812	126	0.0017
12	1 / 4	2	-	-	882	2.71	-	1
13	2/4	1	-	-	942	0.551	-	1
14	2/4	2	-	-	882	1.51	-	1
15	3 / 4	1	-	-	942	0.289	-	1
16	3 / 4	2	-	-	882	0.292	-	1
17	4/4	1	-	-	942	0.289	-	1
18	4/4	2	-	-	882	0.292	-	1

Table 2: Fitness evaluations for designs considering N1.

Step (3) Consider K1:

	E	K1/N1	H	K2/N2	P	V
$\mathbf{x}_{(1)}$	1	1/3	1	2/2	1	1

i) Feasible values of K1 are {1,2,3}. As N1 is set to 3, K1 can not equal 4.

ii)E is the next variable, which relates to the valves hence V is the type of variable to be considered.

iii)All possible combinations involving the three parameters are shown in table 3, where N1 = 3, H = 1, K2/N2 = 2/2 and P=1 remain fixed. Fitness evaluations are also stated, concluding that only the designs in rows 3 and 4 lie in the tolerable slacks for cost and spurious trip frequency.

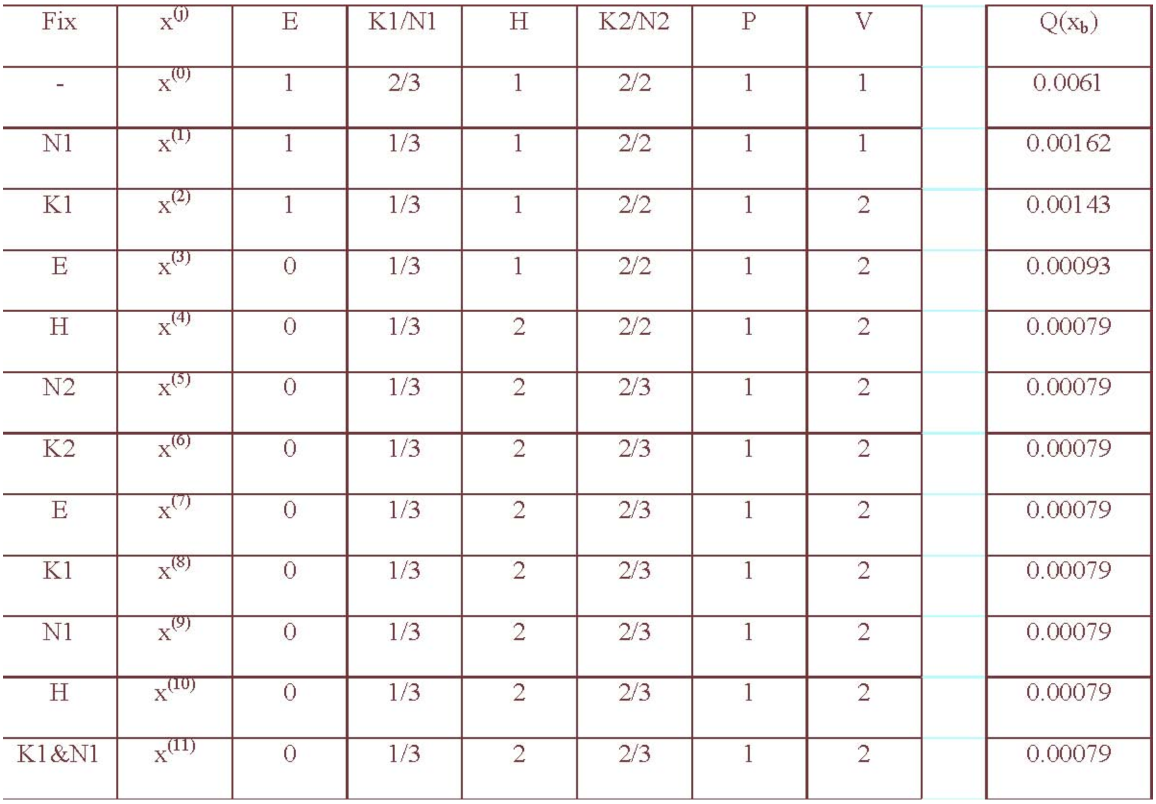
iv)The best design arises in row 4 of table 3, with unavailability of 0.0014. For the next iteration K1 is set to 1 and V to 2. The total design specification is given below.

	E	K1/N1	H	K2/N2	P	V
x(2)	1	1/3	1	2/2	1	2

x(1) K1,E,V	**K1**	**E**	**V**	θ_1	θ_2	**Cost**	F_{SYS}	**MDT**	Q_{SYS}
1	1	0	1	-	-	652	-	-	1
2	1	0	2	-	-	602	-	-	1
3	1	1	1	48	20	922	0.681	130	0.0016
4	1	1	2	48	16	822	0.847	130	0.0014
5	1	2	1	-	-	1192	-	-	1
6	1	2	2	-	-	1042	-	-	1
7	2	0	1	-	-	652	-	-	1
8	2	0	2	-	-	602	-	-	1
9	2	1	1	-	-	922	0.289	-	1
10	2	1	2	-	-	822	0.455	-	1
11	2	2	1	-	-	1192	-	-	1
12	2	2	2	-	-	1042	-	-	1
13	3	0	1	-	-	652	-	-	1
14	3	0	2	-	-	602	-	-	1
15	3	1	1	-	-	922	0.289	-	1
16	3	1	2	-	-	822	0.455	-	1
17	3	2	1	-	-	1192	-	-	1
18	3	2	2	-	-	1042	-	-	1

Table 3: Fitness evaluations for designs considering K1.

Consideration is given to each variable in turn. In total step (3) is carried out eleven times. The modifications to the initial design following each iteration are specified in table 4. Note that Q_{SYS} is the system unavailability of the best design vector, x_b, resulting from an iteration of step 3.



Fix	$x^{(j)}$	E	K1/N1	H	K2/N2	P	V		$Q(x_b)$
-	$x^{(0)}$	1	2/3	1	2/2	1	1		0.0061
N1	$x^{(1)}$	1	1/3	1	2/2	1	1		0.00162
K1	$x^{(2)}$	1	1/3	1	2/2	1	2		0.00143
E	$x^{(3)}$	0	1/3	1	2/2	1	2		0.00093
H	$x^{(4)}$	0	1/3	2	2/2	1	2		0.00079
N2	$x^{(5)}$	0	1/3	2	2/3	1	2		0.00079
K2	$x^{(6)}$	0	1/3	2	2/3	1	2		0.00079
E	$x^{(7)}$	0	1/3	2	2/3	1	2		0.00079
K1	$x^{(8)}$	0	1/3	2	2/3	1	2		0.00079
N1	$x^{(9)}$	0	1/3	2	2/3	1	2		0.00079
H	$x^{(10)}$	0	1/3	2	2/3	1	2		0.00079
K1&N1	$x^{(11)}$	0	1/3	2	2/3	1	2		0.00079

Table 4: Design alterations through execution of step 3.

Step 4: Evaluating the system designs for all possible maintenance test intervals gives the resulting optimal design:

E K1/N1

H

K2/N2

P

V

COST

MDT

θ1

θ2

F_{SYS}

Q_{SYS}

$x^{(11)}$

0 1/3

2

2/3

1

2

40

24

842

0.847

130

7.92x10^{-4}

Step 5: It can be seen from table 3 that iterations 5 to 11 portray consistent predictions of the optimal design.

Corresponding Author: L.M.Bartlett Email: L.M.Bartlett@lboro.ac.uk Fax: 01509 227275 The approach was further tested on four different initial designs. In each case the initial design and final optimal design are stated in table 5. Table 6 specifies the fitness components associated with the optimal design resulting from each start point.

Run No.		E	K1/N1	H	K2/N2	P	V	θ1	θ2	Q_{SYS}	Q'_{SYS}
1	$\mathbf{x}_{(0)}$	1	2/3	1	2/2	1	1	23	36	0.00137	0.0061
	$\mathbf{x}_{(11)}$	0	1/3	2	2/3	1	2	40	24	7.92x10^{-4}	7.92x10^{-4}
2	$\mathbf{x}_{(0)}$	1	4/4	1	3/3	1	1	70	50	0.0094	0.0094
	$\mathbf{x}_{(11)}$	0	1/4	2	2/4	1	2	28	36	8.26x10^{-4}	8.26x10^{-4}
3	$\mathbf{x}_{(0)}$	2	1/1	2	4/4	1	2	40	65	0.0036	0.0067
	$\mathbf{x}_{(11)}$	0	1/1	2	1/2	1	1	28	40	0.001	0.001
4	$\mathbf{x}_{(0)}$	1	1/3	1	2/3	2	2	89	28	0.004	0.014
	$\mathbf{x}_{(11)}$	0	1/4	2	2/4	1	2	28	36	8.26x10^{-4}	8.26x10^{-4}
5	$\mathbf{x}_{(0)}$	2	1/1	1	1/1	2	2	70	50	0.013	0.11
	$\mathbf{x}_{(11)}$	0	1/1	2	1/3	1	2	32	28	8.31x10^{-4}	8.31x10^{-4}

Table 5: Results from additional runs using Logical Search Algorithm

Run No.		Cost	MDT	F_{SYS}	$Q_{SYS} = Q'_{SYS}$
1	$\mathbf{x}_{(11)}$	842	130	0.847	7.92x10$_{-4}$
2	$\mathbf{x}_{(11)}$	882	129.6	0.978	8.26x10$_{-4}$
3	$\mathbf{x}_{(11)}$	882	129.1	0.681	0.001
4	$\mathbf{x}_{(11)}$	882	129.6	0.978	8.26x10$_{-4}$
5	$\mathbf{x}_{(11)}$	802	128.6	0.977	8.31x10$_{-4}$

Table 6: Fitness of optimal designs given in table 4.

The best design is achieved in run number 1 and has a system unavailability of 7.92 x 10^{-4}. The optimal design achieved in each run is highly fit, yet there is significant variety in the parameter set of each optimal design vector. The resulting design is highly dependent on the choice of the initial design point.

7. Ordering of Variables

7.1 Alternative Method 1

The order in which variables are considered in the look-ahead search determines the type of interactions in the parameter set that are analysed. The particular ordering used so far is chosen primarily to focus on interactions between variables in subsystem 1 and interactions between the variables in subsystem 2. The interaction between the number of ESD valves and the number of HIPS valves is also investigated.

The ordering used thus far considers little interaction between the pressure transmitters of each system. An alternative ordering is considered which addresses this. Using the same initial designs as specified in table 5, the results obtained using the alternative ordering of {N1, K1, E, H, N2, K2, E, K1, N1, N2, K2, H} are given in table 7.

Using the alternative ordering, consistently fit designs were achieved in each run as shown in table 8. There is little variety between the optimal design vectors, the main differences arise in the number of pressure transmitters and the number of transmitters required to trip the system. The fittest design arises in run number 5 and has a system unavailability of 7.23 x 10^{-4}.

Run No.	E	K1/N1	H	K2/N2	P	V	θ1	θ2	Q_{SYS}	Q'_{SYS}

1	$\mathbf{x}_{(0)}$	1	2/3	1	2/2	1	1	23	36	0.00137	0.0061
	$\mathbf{x}_{(11)}$	0	1/3	2	2/3	1	2	40	24	7.92x10$_{-4}$	7.92x10$_{-4}$
2	$\mathbf{x}_{(0)}$	1	4/4	1	3/3	1	1	70	50	0.0094	0.0094
	$\mathbf{x}_{(11)}$	0	1/3	2	2/3	1	2	40	24	7.92x10$_{-4}$	7.92x10$_{-4}$
3	$\mathbf{x}_{(0)}$	2	1/1	2	4/4	1	2	40	65	0.0036	0.0067
	$\mathbf{x}_{(11)}$	0	1/2	2	2/3	1	2	40	24	7.92x10$_{-4}$	7.92x10$_{-4}$
4	$\mathbf{x}_{(0)}$	1	1/3	1	2/3	2	2	89	28	0.004	0.014
	$\mathbf{x}_{(11)}$	0	1/3	2	2/3	1	2	40	24	7.92x10$_{-4}$	7.92x10$_{-4}$
5	$\mathbf{x}_{(0)}$	2	1/1	1	1/1	2	2	70	50	0.013	0.11
	$\mathbf{x}_{(11)}$	0	1/2	1	1/2	1	2	34	26	7.23x10$_{-4}$	7.23x10$_{-4}$

Table 7: Results using ordering {N1, K1, E, H, N2, K2, E, K1, N1, N2, K2, H}

Run No.		Cost	MDT	F_{SYS}	$Q_{SYS} = Q'_{SYS}$
1	$\mathbf{x}_{(11)}$	842	130	0.847	7.92x10$_{-4}$
2	$\mathbf{x}_{(11)}$	842	130	0.847	7.92x10$_{-4}$
3	$\mathbf{x}_{(11)}$	822	128.7	0.717	7.92x10$_{-4}$
4	$\mathbf{x}_{(11)}$	842	130	0.847	7.92x10$_{-4}$
5	$\mathbf{x}_{(11)}$	802	129.3	0.977	7.23x10$_{-4}$

Table 8: Fitness of optimal design using alternative ordering method 1.

7.2 Alternative Method 2

Due to the fact that the look-ahead search incorporates only two variables at a time certain combinations of pressure transmitters are never explored. In addition, from the previous results little is achieved by considering variable H a second time. For this reason an ordering is established which considers a three-way iteration between N1, N2 and K2. Hence, method 2 ordering is {N1, K1, E, H, N2, K2, E, K1, N1&N2&K2} where the 9th iteration of step 3 in the Branching Search algorithm considers all possible combinations of N1, N2 and K2.

The same designs are used and the results are given in table 9. Each run results in the same optimal design vector, each has fitness values of: cost = 802 units, MDT = 129.3 hours, and spurious trip frequency = 0.977/year. The design is highly fit with a system unavailability of 7.23x10^{-4}.

Run No.		E	K1/N1	H	K2/N2	P	V	θ1	θ2	Q$_{SYS}$	Q′ $_{SYS}$
1	$\mathbf{x}_{(0)}$	1	2/3	1	2/2	1	1	23	36	0.00137	0.0061
	$\mathbf{x}_{(11)}$	0	1/2	2	1/2	1	2	34	26	7.23x10$_{-4}$	7.23x10$_{-4}$
2	$\mathbf{x}_{(0)}$	1	4/4	1	3/3	1	1	70	50	0.0094	0.0094
	$\mathbf{x}_{(11)}$	0	1/2	2	1/2	1	2	34	26	7.23x10$_{-4}$	7.23x10$_{-4}$
3	$\mathbf{x}_{(0)}$	2	1/1	2	4/4	1	2	40	65	0.0036	0.0067
	$\mathbf{x}_{(11)}$	0	1/2	2	1/2	1	2	34	26	7.23x10$_{-4}$	7.23x10$_{-4}$
4	$\mathbf{x}_{(0)}$	1	1/3	1	2/3	2	2	89	28	0.004	0.014
	$\mathbf{x}_{(11)}$	0	1/2	2	1/2	1	2	34	26	7.23x10$_{-4}$	7.23x10$_{-4}$
5	$\mathbf{x}_{(0)}$	2	1/1	1	1/1	2	2	70	50	0.013	0.11
	$\mathbf{x}_{(11)}$	0	1/2	2	1/2	1	2	34	26	7.23x10$_{-4}$	7.23x10$_{-4}$

Table 9: Results using alternative ordering method 2.

8. Discussion of Results

Using the Branching Search Approach produces designs that are highly fit. An ordering in which the components are considered that accounts for interaction between the design variables is beneficial. The first alternative ordering method (section 7.1) considers a different listing on the second occurrence of each design variable as opposed to merely repeating the variable ordering. Interaction between the pressure transmitters of each system is considered in this way. The second alternative ordering method (section 7.2) considers interaction between a number of variables at once as opposed to simply looking ahead to the next variable. This requires greater computer effort, due to the number of system unavailability evaluations required, which could be exacerbated the more complex the system under consideration, however for this relatively simple system it proves beneficial.

The HIPS system has characteristics which lend themselves well to the Logical Search optimisation procedure. Firstly, the system has only a small number of design variables and the range covered by these is also small. In addition, the variables are segregated into two groups, those concerned with subsystem 1 and those with subsystem 2. There is obviously interaction between the design variables, however the simplicity of the system lends itself well to establishing an efficient order for the variables to be considered in the look-ahead search. These features of the HIPS system are not un-typical of safety systems.

The suitability of any optimisation technique is solely dependent on the specific problem. Techniques that work well for some systems may perform poorly for other systems and vice versa. The main concern for a designer is to find a technique that has the capability to produce a solution, as shown by this technique. The additional length of time that some techniques might take to reach a solution is relatively unimportant compared to the time (and cost) taken to make changes after an incorrect design has been implemented. The Branching Search approach appears to be both highly efficient and accurate, however, its scope of applicability is limited due to its high demand on function evaluations. It is in effect a local approach, which is highly dependent on the choice of initial design vector. This dependency on the start point will increase as the complexity of the optimisation problem increases. In addition, a high degree of knowledge concerning the system is required to establish both the order in which to consider the variables in the look-ahead search and the boundaries used for the tolerable slacks. For these larger problems these complexities would need to be tackled given each problem, however, it is envisaged that one possible method to help eliminate the high dependence of an initial design choice is to combine the technique with a global optimisation technique which would find an effective start point.

9. Conclusions

The Branching Search approach has proven to be effective for the HIPS safety system optimisation. Full use of the available resources has been sought and achieved with a resulting highly fit design with low unavailability. The method shows potential for application to a wider range of problems.

Acknowledgement

The authors would like to thank Dr. R. L. Pattison for her contribution to this paper.

References

[1] Andrews JD. Optimal Safety System Design using Fault Tree Analysis. Proc. IMechE, Vol. 208, 1994, pp123-131.

[2] Rauzy A. New Algorithms for Fault Tree Analysis. Reliability Engineering and System Safety, Vol. 40, 1993, pp203-211.

[3] Sinnamon RM, Andrews JD. Fault Tree Analysis and Binary Decision Diagrams. Proceedings of 1996 Reliability and Maintainability Symposium, Las Vegas, Jan 1996, pp215-222.

[4] Sinnamon RM, Andrews JD. New Approaches to Evaluating Fault Trees. Proceedings of ESREL 95 Conference, June 1995, pp241-254.

[5] Sinnamon RM, Andrews JD. Improved Efficiency in Qualitative Fault Tree Analysis. Quality and Reliability Engineering International, Vol 13, 1997, pp293-298.

[6] Sinnamon RM, Andrews JD. Improved Accuracy in Quantitative Fault Tree Analysis. Quality and Reliability Engineering International, Vol 13, 1997, pp285-292.

[7] Andrews JD, Pattison RL. Optimal Safety System Performance. Proceedings of the Annual Reliability and Maintainability Symposium, Philadelphia, 13-16 Jan 1997, pp76-84.

[8] Pattison RL, Andrews JD. Genetic Algorithms in Optimal Safety System Design. IMechE Proceedings Part E, Journal of Process Mechanical Engineering, Vol. E3, 1999, pp187-197.

[9] Andrews JD, Bartlett LM. Grid-Sampling Optimisation of Safety Systems. Proceedings of the International System Safety Conference, Denver, USA, 2002.

[10] Andrews JD, Bartlett LM. Using Statistically Designed Experiments for Safety System Optimisation. Accepted for publication in IMechE Proceedings Part E, Journal of Process Mechanical Engineering.

[11] Reklaitis GV, Ravindran A, Ragsdell KM. Engineering Optimisation. Methods and Applications. John Wiley and Sons, 1983.

[12] Misra KB, Sharma U. An Efficient Algorithm to Solve Integer Programming Problems Arising in Systems Reliability Design. IEEE Trans Reliability, Vol 40 No 1, 1991, pp81-91.