A study of automatic allocation of automotive safety requirements in two modes: components and failure modes

David Parker, Antoine Godof, Yiannis Papadopoulos, Laurent Saintis

This is the accepted manuscript of an article published in SAE technical papers at https://doi.org/10.4271/2018-01-1076

Abstract

ISO 26262 describes a safety engineering approach in which the safety of a system is considered from the early stages of design through a process of elicitation and allocation of system safety requirements. These are expressed as automotive safety integrity levels (ASILs) at system level and are then progressively allocated to subsystems and components of the system architecture. In recent work, we have demonstrated that this process can be automated using a novel combination of model-based safety analysis and optimization metaheuristics. The approach has been implemented in the HiP-HOPS tool, and it leads to optimal economic decisions on component ASILs. In this paper, first, we discuss this earlier work and demonstrate automatic ASIL decomposition on an automotive example. Secondly, we describe an experiment where we applied two different modes of ASIL decomposition. In HiP-HOPS, it is possible to decompose ASILs either to the safety requirements of components or individual failure modes of components. Protection against independent failure modes could, in theory, be achieved at different ASILs and this will lead to reduced design costs. Although ISO26262 does not explicitly support this option, we have studied the implications of this more refined decomposition on system costs but also on the performance of the decomposition process itself, and we report on the results. Finally, motivated by our study on ASIL decomposition, we discuss the general need for increased automation of safety analysis in complex systems, especially autonomous systems where an infinity of possible operational states and configurations makes manual analysis infeasible.

Introduction

Systems of classification for different levels of safety integrity have been introduced in several different safety standards. While the safety standard IEC 61508 first popularized the Safety Integrity Level (SIL), other safety standards such as ISO 26262 and ARP4754-A developed domain specific versions. The aerospace industry, for example, defines the Development Assurance Level (DAL) in their ARP4754-A standard. ISO 26262, an automotive safety standard [1] defines the Automotive Safety Integrity Level (ASIL) which is the focus of the work in this paper, though the principles are applicable generally across domains.

One of the purposes of the ASIL is to address the issue of traceability with regards to safety in the design of systems. This should be applicable from the early stages of the design process, while initial concepts are being considered, right through to the operational phases of the final product and capture how requirements have been refined and met by the design.

The inevitable and increasing use of software systems in place of purely mechanical systems has meant that traditional techniques of expressing safety requirements as maximum target probabilities for system failures are no longer sufficient.

The ASIL concept is used instead to represent the stringency of safety requirements with respect to software and systematic failures in general. They range from ASIL A (least strict) to ASIL D (most strict). Additionally, QM is used when no special safety requirements are needed indicating only routine Quality Management should be applied.

The elicitation of these safety requirements, as prescribed by the ISO 26262 standard, begins with a hazard and risk analysis to identify potential malfunctions and their hazardous consequences. Based on the severity, likelihood, and controllability of the identified hazards an ASIL is assigned to the hazard to generate the necessary requirements to ensure that any associated risks are reduced to an acceptable level.

Traceability is partially delivered through the process of allocation and decomposition of the ASILs throughout the sub-systems and sub-functions of the system as it is refined from the early concepts. The ISO 26262 standard describes how components that directly cause a hazard receive the ASIL of the hazard. It also lays out guidelines for where multiple components must be involved to cause the hazard. In this instance the components can share the burden of complying with the hazard's ASIL. A process of decomposition (described further later) is defined by the standard to specify what options are allowed when distributing the load of responsibility for meeting a hazards ASIL.

However, the practical application of this decomposition is fraught with difficulty. It requires practitioners to have intricate knowledge of the system being considered including the consequences of architectural failure behavior and how it propagates through the system. This problem is exacerbated by the increases in complexity found in modern systems with more and more interconnected functions delivered through a mix of software and hardware. An explosion of possible operational states, particularly in autonomous systems that are required to work in heterogeneous environments make it even more difficult. The lack of supporting examples in the

ISO 26262 standard is not helpful here and the lack of clarity can often lead to mistakes [2].

A further consideration that is not provided by the standard is that meeting the safety requirement is not the end of the story when it comes to the practical application of the guidance. Coming up with a decomposition of ASILs in a system that satisfies the safety requirements of the identified hazards is a difficult task by itself. However, doing so is in fact merely meeting a constraint. Once that constraint has been met (or in meeting that constraint) it becomes necessary to consider the cost implications of doing so.

Applying different levels of stringency to the safety processes of system development has knock on effects on the cost of said development. The ability to allocate and decompose ASILs in a system in a cost effective (even cost optimal) way further strengthens the need for automated methods.

Various approaches have been made to provide automated assistance to the problem of ASIL decomposition beginning with an exhaustive deterministic method [3], and including optimization approaches such as linear programming [4], exact solvers [5], penalty-based genetic algorithms [6], and Tabu-search [7].

The remainder of the paper will outline a case study that will be used to illustrate the process of modelling a system for ASIL decomposition. It will highlight the need for an automated process for applying the decomposition in a cost optimal way and how to do this using a variation on earlier work [7]. Finally, it will discuss the results of applying the process at different levels of granularity (components versus their failure modes) and the implications of doing so.

Hybrid Braking System Case Study

The effects of the different decomposition techniques will be illustrated on the following example system (in more detail [8-9]) shown in Figure 1. It is designated a 'hybrid' braking system as the braking effort is provided through the combination of electro-mechanical brakes (EMB) and the regenerative energy capture from the in-wheel motors (IWM).

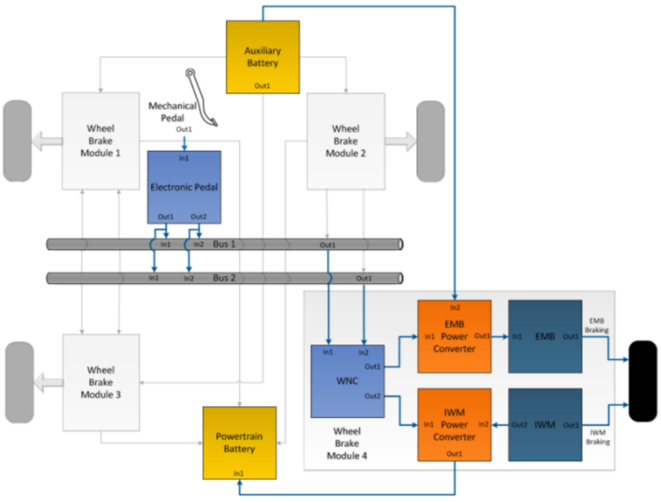

Figure 1. The hybrid-braking system example.

Driver intention is delivered through a mechanical pedal that is sensed and processed through an electronic pedal unit in this brake-by-wire system. The system comprises 4 wheel braking modules, each able to operate independently. In the diagram, wheel brake module 4 shows detail of its components that is matched but not displayed by the other 3 modules. Braking instructions delivered through the redundant duplex communications bus are received by the wheel node controllers (WNC). The WNCs calculate the action required from the wheel's EMB and IWM actuators and deliver the instructions to the respective power converters. The IWM can provide braking functionality by converting the kinetic energy of the vehicle to electric charge which is delivered to the main powertrain battery. This has the benefit of increasing the range of the vehicle, but at high speeds and periods when the battery is in a full state of charge the full braking needs of the vehicle cannot be met. Hence the need for the partnering EMB. The EMB draws power from an auxiliary battery.

In this example, the hazards in Table 1 were identified for the system and, based on the severity of the hazard, the respective ASILs were assigned to them.

Table 1. This table shows the assigned ASILs for the top-level hazards of the system.

#	Hazard	ASIL
1	Incorrect Value Braking	D
2	Loss of Braking Rear Wheels	C
3	Loss of Braking Front Wheels	D
4	Loss of Braking Diagonal Wheels	C
5	Loss of Braking 3 out of 4 Wheels	D
6	Loss of Braking All Wheels	D

The first hazard in the table represents the scenario where a particular value of braking is requested from the system and a different value is delivered. This could result in either excessive or insufficient braking. The remaining hazards in the table represent an omission of braking (i.e. braking is requested and none is delivered) in combinations of one or more of the 4 wheels.

System Modelling

An important part of the methodology being used here is the ability to iterate on the design. To that end, all of the information being used in the process is derived directly from the system model and provides traceability back from the results to the original model.

The topology of the system model has been modelled in Matlab and Simulink. It is provided by the components, their port interfaces, and the connections between them.

The system's failure model is provided by augmenting the topological model with local failure behavior for each of the components. This local failure behavior is added to the model using HiP-HOPS failure expressions. They describe how deviations of output in a component are caused by either an internal failure of the component or through the propagation of failure from elsewhere in the model represented as a deviation of input of the component.

For example, in this case study model, the EMB Power Converter can fail with an omission of output. This can be caused either by an internal omission causing failure (OFailure) or by an omission deviation of either of its two inputs.

Omission-Out = Omission-In1 or Omission-In2 or OFailure

In contrast the WNC component has two outputs. Each of them can fail by omission, but this deviation of output is either caused by a specific internal failure (OFailure1 and OFailure2 respectively) or by the combination of an omission deviation at both of the inputs.

Omission-Out1 = (Omission-In1 and Omission-In2) or OFailure1

Omission-Out2 = (Omission-In1 and Omission-In2) or OFailure2

Note that at this stage, the system is under design so the precise internal electrical/mechanical/functional component failures are not known. However, the design intention is known and therefore what constitutes potential output failures and their intended relationship to input failures is known. Beyond this, one can hypothesise that each output failure can be caused by one yet unspecified collective internal cause. It is precisely these requirements for avoidance and containment of these internal causes that the decomposition exercise tries to establish via analysis of propagation and effects of those causes of failure. Each failure expression describes a mini-fault tree and each of the components in the system may have one or more to describe how the component propagates, generates, or mitigates failure that it is presented with.

Any deviations of output are propagated through the connections in the model to the inputs of the connected components. In the example, the first output of the WNC is connected to one of the inputs of the EMB Power Converter. Matching failure classes (e.g. Omission) found at either end of such a connection allow the mini-fault trees to be joined.

For example, the omission of the second input of the EMB Power Converter can be replaced by the expression for the omission of the first output of the WNC.

Omission-EMBPC.Out = Omission- EMBPC.In1 or **(Omission-WNC.In1 and Omission-WNC.In2) or WNC.OFailure1** or EMBPC.OFailure

The part of the expression that relates to the WNC is shown above in bold and additional identifiers have been added to indicate which component the ports and failures originate in.

This process of combining the mini-fault trees of the components begins at the hazards that have been identified for the system. These are connected to the outputs of the systems using the same Boolean expressions. For example, the hazard “Loss of braking of all wheels” is connected using the following expression:

Omission-Brake_Unit1.Braking and
Omission-Brake_Unit2.Braking and
Omission-Brake_Unit3.Braking and
Omission-Brake_Unit4.Braking

Each braking units’ omission of output as a failure expression that refers to an omission of both the EMB and the IWM function, and so on. This process of combining the mini-fault trees of the components in the model continue until all of the connected input deviations have been resolved.

The result is a complete fault tree that is generated for each hazard defined for the system. The fault tree describes the propagation of failure from the internal failures of the components (the basic events are the leaf nodes of the tree) to the top-level hazards of the system through the combination of Boolean logic.

To be used for the ASIL decomposition process it is necessary to have the fault propagation in it minimal form. This is provided through the automatic fault tree analysis capabilities of the HiP-HOPS engine and results in a set of minimal, non-redundant, cut sets.

For the case study example this results in 6 fault trees (one for each of the hazards), each of which shares branches with the others. Consequently, the cut sets that are generated as the result of the fault tree analyses will be shared across multiple hazards. Table 2 shows the number of minimal cut sets generated for each of the hazards.

Table 2. This table shows the number of minimal cut sets for each of the top-level hazards of the system.

#	Hazard	Cut Sets
1	Incorrect Value Braking	1302
2	Loss of Braking Rear Wheels	103
3	Loss of Braking Front Wheels	103
4	Loss of Braking Diagonal Wheels	202
5	Loss of Braking 3 out of 4 Wheels	3136
6	Loss of Braking All Wheels	6727
	total	11573

The cut sets are important for the ASIL decomposition process as each minimal cut set gives a combination of failure modes that is both necessary and sufficient to cause the hazard. For example, one of the cut sets of the “Loss of Braking Rear Wheels” hazard is an internal omission causing failure of both the auxiliary battery and the powertrain battery.

In particular, the cut sets of order 2 or more (non single points of failure) derived directly from the model show the subsystem independence that is required for decomposition.

The ASIL that has been assigned to this hazard is C. In order to satisfy the safety requirements of the system for this cut set, the ASIL of each of the failures in this cut set could be developed to ASIL C also. However, the ASIL decomposition described in ISO 26262 allows for the allocation of reduced stringency where independent redundancy can be shown. In this case, because the failure of both the powertrain and the auxiliary battery is required to cause the specified hazard, the stringency of the ASIL allocated to each of these failures can be reduced according to the given algebra.

$$\sum_{j=1}^{i} ASIL_{component_j} \geq ASIL_{hazard} \quad (1)$$

To facilitate this, each of the ASILs can be represented by an integer value 0 to 4 as shown in table 3.

Table 3. This table shows the algebraic value for each ASIL.

ASIL	Algebra Value
QM	0
A	1
B	2
C	3
D	4

Table 4 shows all the combinations of ASILs that could be decomposed to the powertrain and auxiliary battery failures respectively along with the algebraic values for each of those ASILs. The final column shows the sum value of the two algebraic values. Where the sum value equals or exceeds 3 (the algebra value associated with ASIL C) the decomposition is deemed to be valid.

The last 6 combinations have a sum value of less than three so can be discarded as invalid decompositions. The four shaded rows show the combinations that exactly meet the requirement. The remaining 15 rows also exceed the stringency of the safety requirement. These can be considered a valid decomposition, however it is likely to be suboptimal once cost is considered as generally delivering a function at a higher safety integrity level is more costly.

Table 4. This table shows the ASIL algebra of possible choices for decomposition due to the powertrain and auxiliary battery cut set for the "loss of braking rear wheels" hazard. The shaded area shows the configurations that exactly meet the requirement.

Powertrain battery		Auxiliary battery		sum
ASIL	algebra	ASIL	algebra	
D	4	D	4	8
D	4	C	3	7
C	3	D	4	7
D	4	B	2	6
B	2	D	4	6
C	3	C	3	6
D	4	A	1	5
A	1	D	4	5
C	3	B	2	5
B	2	C	3	5
D	4	QM	0	4
QM	0	D	4	4
C	3	A	1	4
A	1	C	4	4
B	2	B	2	4
A	1	B	2	3
B	2	A	1	3
C	3	QM	0	3
QM	0	C	3	3
A	1	A	1	2
B	2	QM	0	2
QM	0	B	2	2
A	1	QM	0	1
QM	0	A	1	1
QM	0	QM	0	0

If we consider this one hazard, then we can be satisfied that if any of the shaded combinations from the table are chosen, then we will meet the requirements of avoiding the hazard. However, the reality is more complicated.

This cut set is shared across multiple hazards. One of these is the "Loss of Braking All Wheels" hazard that was assigned ASIL D. When we include this constraint, the shaded combinations are no longer valid as their sum value is less than 4, the algebraic value for ASIL D.

There are 5 combinations that exactly meet the ASIL D requirement, but further factors need to be considered before making a final selection.

The cut set under consideration is of order 2 and contains the failure of the auxiliary battery and the powertrain battery. The auxiliary battery failure is part of an additional 9 order 3 cut sets of the "Loss of Braking Rear" hazard.

As an example we can consider one of these cut sets: omission failure of the auxiliary battery and the IWM of brake unit 3 and the IWM of brake unit 4. The decomposition that we choose for this cut set is affected by the choice of decomposition from the previous cut set. If we chose to allocate ASIL D to the auxiliary battery in the previous cut set, then we could potentially allocate QM to each of the omission failures of the IWMs of brake unit 3 and 4. However, if we had chosen one of the other decompositions such as QM to the auxiliary battery (and ASIL D to the powertrain battery), then the stringency of the decompositions to the other failures in the second cut set would have needed to be higher to meet the requirements.

If we also consider the "Loss of Braking All Wheels" hazard that adds another 81 order 5 cut sets. Then it is necessary to also consider the auxiliary battery's contribution to 3 other hazards and all of their cut sets. Similarly, the choice of decomposition to the first cut set pair also has knock on effects for any and all cut sets that contain the powertrain battery.

The ASIL algebra provides a way of determining the validity of a given decomposition. There are however additional factors that will influence the choice of ASIL combinations when decomposing in a system. A significant one is development cost. The ASIL allocated to a component represents the stringency of requirements that need to be complied with when developing it. Therefore, the higher the ASIL the higher the development cost. Where the safety requirements can be met, it is desirable to find an decomposition of ASILs that minimizes the cost of doing so.

It is often the case in the early stages of the design process, that the precise development costs of the components or functions in a system cannot be provided. That does not mean that cost cannot be considered as part of the decomposition process. In lieu of individual component costs, it is possible to consider the relative expected cost

of development. In the simplest case, the algebra values in table 3 can be used as a linear cost, but this doesn't serve in further distinguishing the different combinations of decompositions. Table 5 provides a non-linear cost heuristic based on the experiential observation that the difference in cost between ASIL B to ASIL C is greater than the difference between the other ASILs [10].

Further exploration of the application of different cost heuristics to the optimization of ASIL decomposition can be found here [11].

Table 5. This table shows the experiential cost heuristic for each ASIL.

ASIL	Cost
QM	0
A	10
B	20
C	40
D	50

When you apply this cost heuristic to the decomposition combinations available for the auxiliary battery and powertrain battery cut set in the "Loss of Braking Rear Wheels" hazard you get the results shown in Table 6. According to the heuristic, the shaded combinations are less costly than the unshaded combinations despite both meeting the safety requirement for that hazard.

Table 6. This table shows the estimated cost of possible choices for decomposition due to the powertrain and auxiliary battery cut set for the "loss of braking rear wheels" hazard.

Powertrain battery		Auxiliary battery		total cost
ASIL	cost	ASIL	Cost	
A	10	B	20	30
B	20	A	10	30
C	40	QM	0	40
QM	0	C	40	40

It is important to remember that the failures in the cut sets are shared in multiple cut sets across multiple hazard fault trees. Therefore, in order to calculate the cost for the system it is summed once per failure and not once for every occurrence in the cut sets.

It is clear that achieving valid ASIL decompositions at minimal cost across a system manually is a practically impossible task. The many possible combinations, the multiple constraints provided by the hazards, and the knock-on effect of the interconnected fault trees and cut sets, leads to a combinatorial explosion. This is an optimization problem that requires the use of automated optimization algorithms to solve.

To do this it is necessary to encode potential solutions to the ASIL decomposition problem. The problem that is to be solved is: what is the ASIL requirement of each component/failure mode in the system such that the requirements assigned to the hazards are satisfied; and at minimum cost. The encoding that the algorithm can work with is a list of all the unique failure modes in the system and the ASIL that has been allocated to it. An example of this is shown in Figure 2.

Solution^t	FM1	FM2	FM3	FM4	FM5
	B	D	C	B	A

Figure 2. Example solution encoding at iteration t showing the ASILs allocated to each unique failure mode in the system.

An encoding can be validated against the hazards' ASILs by considering each cut set in turn, summing the ASIL algebra values of each of the failure modes from the cut set (as provided by the encoding list) and noting whether the sum result is equal to or exceeds the value of the current cut set's hazard ASIL. If this is true for all of the cut sets of all of the hazards, then the current encoding is valid.

The cost of a solution is calculated by looking up the cost (such as in Table 5) of each allocated ASIL in the encoding and summing them together to provide the total ASIL related costs of the system. The example shown in Figure 2 has the ASIL cost of 140 (20 + 50 + 40 + 20 + 10).

Tabu search

The optimization technique applied for this paper uses a Tabu search variant algorithm [7]. It is based on the Steepest Ascent Mildest Descent (SAMD) method used by Hansen and Lih [12] for their work on system reliability optimization. One modification made for the ASIL decomposition problem is to adapt the method to a Steepest Descent Mildest Ascent (SDMA) as the algorithm seeks to minimize the development costs associated with the safety requirements, rather than the maximization objective of the SAMD approach.

The SDMA method attempts to follow the steepest descent path through the search space until a local minimum is detected. In order to escape the local minima, the algorithm uses the mildest ascent route available to it.

In order to achieve the steepest descent during an iteration of the algorithm it is necessary to choose a failure mode from the encoding and reduce its decomposed ASIL by one (i.e. from ASIL C to ASIL B). The reduction in the chosen failure mode's ASIL should result in the largest reduction in system cost. In the case of the example shown in Figure 2 the chosen failure mode would be FM3. The cost difference of reducing from ASIL C to ASIL B is 20, whereas the cost difference of all of the other available reductions is 10 (ASIL D to C, ASIL B to A, and ASIL A to QM as given by the cost heuristic in Table 5). The resultant encoding is shown in Figure 3.

Solution^{t+1}	FM1	FM2	FM3	FM4	FM5
	B	D	**B**	B	A

Figure 3. Example solution encoding at iteration t+1 where the steepest descent was followed by reducing the ASIL of FM3, shown in bold.

To demonstrate selecting the mildest ascent we will assume that the solution at iteration t+1 in Figure 3 is in a local minimum. This can occur if it is not possible to reduce any of the ASILs in the solution without invalidating one or more of the hazards' safety requirements. To produce the mildest ascent, it is necessary to choose one of the failure modes and increase the ASIL of its safety requirement by one such that it results in the smallest increase in cost. In the case of our example we would select FM5 resulting in an increase in cost of 10.

The other choices either result in an increase in cost of 20 (FM1, FM3, and FM4 from ASIL B to ASIL C) or cannot be increase further (FM2 ASIL D). The resultant encoding is shown in Figure 4.

Solution^{t+2}	FM1	FM2	FM3	FM4	FM5
	B	D	B	B	**B**

Figure 4. Example solution encoding at iteration t+2 where the mildest ascent was followed by increasing the ASIL of FM5, shown in bold.

An adaptive memory structure (the Tabu list) is used to prevent the algorithm from making reverse moves and falling back in to local minima. A variable f_i (where i refers to the failure mode that was just increased) stores how many iterations a reverse move will be forbidden for. After making an ascent move this variable is set to a number of iterations *p*. Conversely, following a descent move, the variable f'_i is set to a number of iterations *p'* and stores for how many iterations the failure mode will be blocked from increasing.

The use of such a memory structure increases the diversity in the search by forcing the algorithm to be more explorative. In order to decrease the algorithms sensitivity to the initial selection of the *p* and *p'* values, they are adjusted dynamically, incrementing at intervals $updatePeriod_p$ and $updatePeriod_{p'}$ respectively. When they reach their maximum values $limit_p$ and $limit_{q'}$, they are reset to zero.

The algorithm includes an aspiration criterion which allows it to make a move forbidden by the memory structure if the resultant solution will be superior to any found previously.

Figure 5 summarizes the SDMA Tabu search algorithm used in this paper.

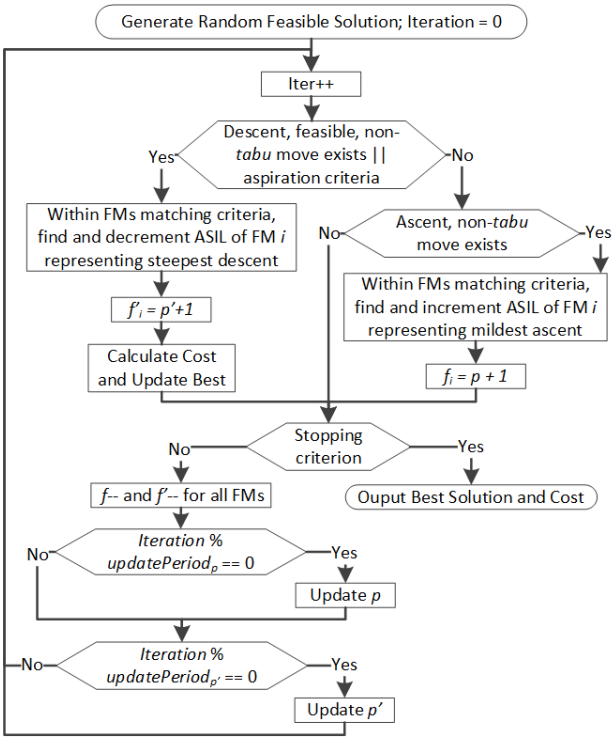

Figure 5. Tabu search overview.

Failure Modes versus Components

Earlier work with the HiP-HOPS ASIL decomposition techniques used the Tabu search algorithm as described in this paper. The encoding for the search algorithm stores an ASIL value for each of the failure modes in the system. It is theorized that taking advantage of the automatic fault tree analysis at the granularity of the failure modes allows for the specification of safety requirements for the development of (sub-) systems and their components that would be superior (less costly) than if forced to allocate at the component level. This approach considers ASIL decomposition at a level that is not described in the ISO 26262 standard, which speaks only of decomposing down to the level of component.

This paper takes a closer look at the consequences of such a limit in terms of the solutions possible when decomposing down to the failure modes as compared to different approaches for achieving this at the component level.

The first approach being considered is a naïve conversion. This involves running the previous ASIL decomposition algorithm to allocate ASILs to the failure modes of the system. The failure modes can be traced back to the system model that generated them. This means that for each component in the system, it is possible to collect the highest ASIL that was decomposed to one of its failures.

For example, in the HBS case study, the auxiliary battery component has two failure modes: an omission and a value failure. As a result of the optimization algorithm, they are allocated ASIL B and ASIL D respectively. Selecting the highest of these values results in us allocating ASIL D to the auxiliary battery component.

The second approach involves altering the optimization algorithm so that the encoding of the solution is not a list of ASILs decomposed to each of the failure modes in the system, but rather at the less granular level of the components. The algorithm manipulates the allocated ASILs in the encoding in the same manner as before. However, in order to establish the validity of the decomposition it is necessary to associate the components ASIL with all of its failure modes. These in turn are then used to validate the decompositions through the cut sets as before.

An example of this would be that if the ASIL allocated to the auxiliary battery in the solution encoding was ASIL C, then both the omission failure and the value failure of that component would be set to ASIL C. The validity check would reveal this to be an invalid decomposition due to one or more of the hazards' ASIL requirement.

To calculate the development cost in both approaches to optimizing the ASIL decomposition at the component level, the value is calculated by summing the heuristic cost of the effective component faults' ASILs. For example, the cost of setting the auxiliary battery ASIL to C is 80 because it has two failure modes that both derive their ASILs from the parent component. It is done this way for this paper so that the resultant cost can be directly compared from all three approaches.

Results

Table 7 shows the results of running the optimization algorithm in each of the three approaches. The first column indicates the 60 unique failure modes in the Hybrid braking case study. The naming convention used here gives first the name of the component followed

by the name of the failure mode separated by a period. For example, EMB1.Omission refers to the omission failure of the electro-mechanical brake in the first wheel brake module.

The second (Hazard) column shows the hazards (indexed in table 1 with its ASIL) that the failure mode contributes to through the (many) cut sets (shown in table 2). In all cases the failure mode contributes (at least indirectly) to a hazard with ASIL D. The third (FM) column shows the ASIL that is allocated using the pure direct to failure mode optimization approach. The fourth (FM->C) column shows the ASIL that are derived from assigning the highest ASIL from the first approach to the parent component of each of the failure modes. The final column (C) shows the ASIL that is allocated when the optimization algorithm decomposes the ASILs directly to the components of the system.

At the bottom of the table the ASIL development cost is noted for each of the three approaches.

The shaded cells in the last column highlight where the allocations made by the two different component focused algorithms are different.

Table 7. This table shows the decomposed ASILs for the failure modes of the system when using the different decomposition techniques. The FM column shows the original HiP-HOPS technique that decomposes to the failure modes in the system. The FM->C column post-processes the ASILs to assign the highest sub-value to each component. The C column optimizes directly to the components. The cells marked in grey highlight differences between the latter two results.

		ASILs allocated per:		
Failure Mode	**Hazard**	**FM**	**FM -> C**	**C**
Battery_Aux.Omission	2-5	B	D	D
Battery_Aux.Value	1	D	D	D
Battery_PT.Omission	2-5	B	B	B
Battery_PT.Value	1	B	B	B
Communication_Bus1.Omission	2-5	B	D	D
Communication_Bus2.Omission	2-5	B	D	D
Electronic_Pedal.Omission1	1-6	D	D	D
Electronic_Pedal.Omission2	2-5	B	D	D
Electronic_Pedal.Value1	1	D	D	D
Electronic_Pedal.Value2	1	D	D	D
Mechanical_Pedal.Omission	1-6	B	D	D
Mechanical_Pedal.Value	1	D	D	D
EMB1.Omission	3-6	QM	QM	B
EMB1.Value	1	QM	QM	B
EMB1_Power_Converter.Omission	3-6	QM	QM	B
EMB1_Power_Converter.Value	1	QM	QM	B
IWM1.Omission	3-6	A	A	B
IWM1.Value	1	QM	A	B
IWM1_Power_Converter.Omission	3-6	QM	A	B
IWM1_Power_Converter.Value	1	A	A	B
WNC1.Omission1	3-6	QM	A	B
WNC1.Omission2	3-6	A	A	B
WNC1.Value1	1	QM	A	B
WNC1.Value2	1	QM	A	B
EMB2.Omission	3-6	B	B	QM
EMB2.Value	1	B	B	QM
EMB2_Power_Converter.Omission	3-6	B	B	QM
EMB2_Power_Converter.Value	1	B	B	QM
IWM2.Omission	3-6	A	B	QM
IWM2.Value	1	B	B	QM
IWM2_Power_Converter.Omission	3-6	B	B	QM
IWM2_Power_Converter.Value	1	A	B	QM
WNC2.Omission1	3-6	B	B	QM
WNC2.Omission2	3-6	A	B	QM
WNC2.Value1	1	B	B	QM
WNC2.Value2	1	B	B	QM
EMB3.Omission	2,4-6	A	A	B
EMB3.Value	1	A	A	B
EMB3_Power_Converter.Omission	2,4-6	A	A	B
EMB3_Power_Converter.Value	1	A	A	B
IWM3.Omission	2,4-6	QM	A	B
IWM3.Value	1	A	A	B
IWM3_Power_Converter.Omission	2,4-6	A	A	B
IWM3_Power_Converter.Value	1	QM	A	B
WNC3.Omission1	2,4-6	A	A	B
WNC3.Omission2	2,4-6	QM	A	B
WNC3.Value1	1	A	A	B
WNC3.Value2	1	A	A	B
EMB4.Omission	2,4-6	A	A	QM
EMB4.Value	1	A	A	QM
EMB4_Power_Converter.Omission	2,4-6	A	A	QM
EMB4_Power_Converter.Value	1	A	A	QM
IWM4.Omission	2,4-6	A	A	QM
IWM4.Value	1	A	A	QM
IWM4_Power_Converter.Omission	2,4-6	A	A	QM
IWM4_Power_Converter.Value	1	A	A	QM
WNC4.Omission1	2,4-6	A	A	QM
WNC4.Omission2	2,4-6	A	A	QM
WNC4.Value1	1	A	A	QM
WNC4.Value2	1	A	A	QM
Total cost		840	1100	1020

What these results show is that the finer granularity of allocating down to the level of the failure modes allows the algorithm to find a more cost-effective solution. This would seem to be highly desirable in situations where vendors would be able to deliver components that can meet specific safety requirements for the different failure modes of the parts, or where the component is effectively a subsystem and adequate partitioning can be established between elements within.

Where this is not possible it is necessary to specify the safety requirements at the level of the components, which is more in keeping with the process as laid out by the ISO 26262 standard. Here, using the failure mode allocation technique and converting the results to the component level produces and inferior, less cost-effective

solution that optimizing directly to the components using the specialized algorithm.

In the latter approach the components of each wheel brake module are treated more uniformly and because they represent independent redundancy the distribution of the ASILs is more favorable.

Table 8. This table shows an alternative logarithmic cost heuristic for each ASIL.

ASIL	Cost
QM	0
A	10
B	100
C	1000
D	10000

This is not the end of the story however as the ability of the direct to component allocation algorithm to find superior solutions to the conversion approach depends on the cost heuristic being used. If, for example, a logarithmic cost heuristic is used like that in Table 8, then the solution identified by the two component focused approaches is the same. This is shown in Table 9.

Table 9. This table shows the costs of running the different optimization approaches with a logarithmic cost heuristic such as in Table 8.

ASILs allocated per:		
FM	**FM -> C**	**C**
51150	100680	100680

In order for the direct to component optimization to find superior solutions, it is necessary for the cost heuristic to have moves between different ASILs to have interchangeable cost differences. For example, with the cost heuristic shown in Table 5 only the jump from ASIL B to ASIL C is unique (20 units compared to 10 for all the other jumps). The logarithmic cost heuristic in Table 8 has unique cost jumps for all of its ASILs. It should be noted that the direct to failure mode approach finds markedly superior solutions in all cases.

An additional consideration is the performance cost. When optimizing directly to the components the search space is considerably reduced. There are 60 failure modes in the case study system but only 24 components. The direct to components algorithm took a little over a second to complete one run of the algorithm compared to just under 9 seconds to run the direct to failure modes algorithm.

With these different factors in consideration it appears that the obvious choice when constrained to consider ASILs at a component level only is to use an algorithm that specially targets that objective directly. It is quicker, and the resultant configuration of ASIL allocations may be superior.

However, if it is possible to consider the allocation of ASILs to the more granular level of the failure modes of a system, then a more cost effective solution is likely to be found.

Conclusions

The safety engineering approach described in the automotive standard ISO 26262 requires the consideration of safety right from the early stages of the design process. One of the key pillars of this are the ASILs that can be assigned to the safety requirements of the system. Importantly, these requirements can then be distributed throughout the components of the system and decomposed where independent redundancy can be shown to manage the cost of meeting these requirements.

There is additional effort/cost required due to decomposition (for example, proof of independence needed) which isn't considered in this study. This cost likely not negligible and it would be worth estimating these costs in the future. However, decomposition is precisely used in order to reduce costs so the relative cost of decomposition in general must be significantly lower than the benefits of reducing ASILs.

Doing this manually, even in small systems is impractical to the point of being impossible if the expectation is to achieve cost optimality. Automated systems are necessary to cover the vast search spaces that are generated by the combinatorial explosion of potential configuration.

This paper described the recent work in this area implemented in the HiP-HOPS safety analysis and optimization tool. Two modes of operation are shown, allocation to components as intended by the ISO 26262 standard, and the theoretical allocation down to the level of component failure modes.

The approach described here is not a 'fire and forget', one-time application to provide automatic safety standard compliance. Rather it should be considered as an assistive technique to help inform engineer choices in their efforts for cost-effective standard compliance; one that can be applied iteratively throughout the design life of a system.

Comparison of the two modes reveals the economic benefits available where we are able to use the latter, more granulated allocation process. Where this is not possible specialized component focused algorithms offer potential advantages over simply converting the results. In all cases, it is more efficient working with a smaller search space, and in some cases may provide superior, more cost effective solutions, though this will depend on the cost heuristic being used.

References

1. Int'l Organization for Standardization, "ISO 26262 Road vehicles - Functional safety," 2011
2. Ward, D.D. and S.E. Crozier. The uses and abuses of ASIL decomposition in ISO 26262. in System Safety, incorporating the Cyber Security Conference 2012, 7th IET International Conference on. 2012.
3. Y. Papadopoulos, M. Walker, M-O. Reiser, D. Servat, A. Abele, R. Johansson, H. Lonn, M. Torngren, M. Weber (2010) Automatic Allocation of Safety Integrity Levels, 8th European Dependable Computing Conference - CARS workshop, Valencia, Spain, Spain, pp. 7-11, ACM press, ISBN:978-1-60558-915-2
4. Mader, R., Armengaud, E., Leitner, A., & Steger, C. (2012). Automatic and Optimal Allocation of Safety Integrity Levels.

Reliability and Maintainability Symposium (RAMS 2012), (pp. 1-6). Reno, NV, USA. doi:10.1109/RAMS.2012.6175431
5. Murashkin, A., Silva Azevedo, L., Guo, J., Zulkoski, E. et al., "Automated Decomposition and Allocation of Automotive Safety Integrity Levels Using Exact Solvers," SAE Int. J. Passeng. Cars – Electron. Electr. Syst. 8(1):70-78, 2015, https://doi.org/10.4271/2015-01-0156.
6. D. Parker, M. Walker, L. Azevedo, Y. Papadopoulos, R. Araujo (2013) Automatic Decomposition and Allocation of Safety Integrity Levels using a Penalty-based Genetic Algorithm. Proceedings of the 26th International Conference on Industrial, Engineering, and other Applications of Applied Intelligent Systems (IEA/AIE 2012): Special session on Decision Support for Safety-Related Systems. 17-21st June, Amsterdam, The Netherlands.
7. L. Silva Azevedo, D. Parker, M. Walker, Y. Papadopoulos, and R. Esteves Araujo (2013) Automatic Decomposition of Safety Integrity Levels: Optimisation by Tabu Search. 2nd Workshop on Critical Automotive applications: Robustness & Safety (CARS), at the 32nd International Conference on Computer Safety, Reliability, and Security (SAFECOMP'13), Toulouse, France, 2013.
8. Azevedo L.P. (2012) Hybrid Braking System for Electrical Vehicles: Functional Safety, M.Sc. thesis, Dept. Elect. Eng., Porto Univ., Porto, Portugal, 2012.
9. R. de Castro, R. E. Araújo, and D. Freitas. (2011) "Hybrid ABS with Electric motor and friction Brakes," presented at the IAVSD2011 - 22nd International Symposium on Dynamics of Vehicles on Roads and Tracks, Manchester, UK, 2011.
10. Allen, M. (2012).: Cost Versus ASIL. ISO 26262 Functional Safety [LinkedIn]. 2 February 2012. Available at: http://www.linkedin.com/groups/Cost-versus-ASIL-2308567.S.92692199?view=&srchtype=discussedNews&gid=2308567&item=92692199&type=member&trk=e ml-anet_dig-b_pd-ttl-cn&ut=1evtvoEm1QcBw1. [Accessed 1 May 14].
11. Azevedo L.S., Parker D., Papadopoulos Y., Walker M., Sorokos I., Araújo R.E. (2014) Exploring the Impact of Different Cost Heuristics in the Allocation of Safety Integrity Levels. In: Ortmeier F., Rauzy A. (eds) Model-Based Safety and Assessment. Lecture Notes in Computer Science, vol 8822. Springer, Cham
12. Hansen, P. and Lih, K.-W. (1996): Heuristic reliability optimization by tabu search. Annals of Operations Research, volume (63), pp. 321-336

Contact Information

David Parker d.j.parker@hull.ac.uk

University of Hull, UK