A Systems Reliability Approach to Decision Making in Autonomous Multi-Platform Systems Operating a Phased Mission

John D. Andrews, Loughborough University

Darren R. Prescott, Loughborough University

Rasa Remenyte-Prescott, Loughborough University

Key Words: phased mission, autonomous system, fault tree, binary decision diagram, multi-platform system

SUMMARY & CONCLUSIONS

This paper presents a decision making strategy for autonomous multi-platform systems, wherein a number of platforms perform phased missions in order to achieve an overall mission objective. Phased missions are defined for both single and multi-platform systems and a decision making strategy is outlined for such systems. The requirements for a tool performing such a strategy are discussed and methods and techniques, traditionally used for system reliability assessment, are identified to fulfill these requirements. Two examples are presented in order to demonstrate how a decision making tool would be employed in practice. Finally, a brief discussion of the efficient implementation of such a strategy is presented.

1 INTRODUCTION

Many systems perform missions that consist of several distinct phases carried out in sequence. In order for the mission to be successful, each of the phases must be completed successfully. The requirements for the successful completion of each phase are different as, therefore, are the causes of failure in each phase. An example of such a phased mission is an aircraft flight. The simple flight can be considered as involving phases: taxi to the runway, take off, climb to a cruising altitude, cruise, descend, land and taxi back to the terminal. Other examples of phased missions include military operations of aircraft or ships and spacecraft missions. The consequences of failure of such missions are often high and, as such, accurate methods of analysis are required to examine failure probabilities in each phase of the mission and of the mission as a whole. Methods used to model the failure probability of such phased missions are fault tree analysis [1], cause-consequence analysis [2] and binary decision diagrams [3].

Autonomous systems can operate in environments which would be of high risk for human operators. Unmanned aerial vehicles (UAV), other unmanned vehicles (UXV), such as land vehicles, or other autonomous robots are examples of such systems. In such autonomous platforms there is a requirement placed on the system to make decisions, without human input, at various stages within the mission as to the next suitable course of action. These decisions must be made in an informed manner, taking into account the risk associated with any actions to the platform itself and also the risk to human life or any objects within the platform's locality.

Increasingly, within many environments, there will be requirements placed upon autonomous platforms to work collaboratively in order to achieve an overall mission goal. An obvious example of this is the progression of some military operations towards network-enabled capability (NEC) or network-centric warfare (NCW). Individual platforms within such multi-platform phased missions must make decisions not only accounting for their own particular phased mission but also accounting for the overall mission being carried out by the group of platforms.

In the cases of both single and multi-platform phased missions, the decision making process used by the autonomous platforms is of the utmost importance. A key factor in making decisions about whether or how to implement future mission phases for individual platforms will be the likelihood of failure of the platform, or of other platforms, within future mission phases. Components can fail and external conditions can change once the mission gets underway. There are two distinct times when a mission prognosis can be provided and used to make an informed decision on the future of the mission. Before a mission is due to start a prognosis can be used to determine whether or not the mission should begin in its current configuration. Once the mission is underway a prognosis can be made as the conditions change. This would take into account that certain parts of the mission have been completed and also if failures have occurred on platforms, systems, subsystems or components. If the prognosis of mission success falls below pre-determined, acceptable levels then the mission could be aborted or a reconfiguration could take place. Such a reconfiguration might involve different platforms within the multi-platform phased mission performing tasks in a different order or performing an altogether new set of tasks. It might even be decided that a whole new mission objective should be considered. Clearly, in a rapidly changing mission environment there is a need for a prognostics tool to provide information quickly and accurately in order that well-informed

1-4244-1461-X/08/$25.00 ©2008 IEEE

decisions can be made in the shortest possible timeframe.

In this paper an overall strategy of using a prognostics capability within a decision making framework is introduced. A method of using established techniques of phased mission reliability analysis to provide a prognostics capability is described for single platform phased missions and this method is then extended to cover multi-platform phased missions.

2 PHASED MISSION DEFINITIONS

Single and multi-platform phased missions have a number of features and characteristics. The assumptions made about each type of phased mission are detailed in the following sections.

2.1 *Single Platform Phased Mission Definition*

A representation of a single platform phased mission with *n* phases is shown in Figure 1.

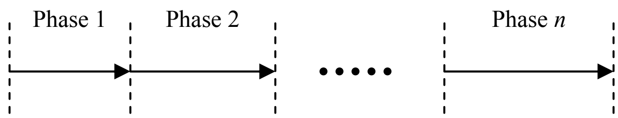

Figure 1 – A Single Platform Phased Mission.

In general a single platform phased mission has the following characteristics:

- A mission consists of a number of consecutive, sequential phases.
- Each phase has different functional requirements and therefore different failure criteria from other phases.
- Mission success requires all phases to be successfully completed.

For the techniques described in this paper a number of further assumptions are made:

- The length of each phase is known.
- The mission is non-repairable and therefore component failures remain in the system once they have occurred.
- All components are in the working state as the mission begins.

2.2 *Multi-Platform Phased Mission Definition*

A multi-platform phased mission is performed by a number of single platforms, each of which performs a phased mission as outlined in section 2.1. The multi-platform phased mission as a whole has the following characteristics:

- Each platform performs a task or number of tasks (defined here as *mission tasks*) that contribute to the overall mission objective as part of its own phased mission.
- The mission tasks are not necessarily sequential – different mission tasks can be carried out in parallel by different platforms.
- Mission tasks can have certain requirements, such as a need for mission tasks to be carried out in strict sequence or to start or end simultaneously.

A number of assumptions are also made:

- For the individual platforms the assumptions are outlined in the previous section.
- The mission is assumed to fail if any of the individual platforms performing the mission fail at any point during the mission. Note that in this case a new mission configuration or objective could be implemented with the remaining platforms. The failure is only of the original mission.

3 OVERALL DECISION MAKING STRATEGY

Here an overall strategy for the formation of a decision making process for phased missions involving autonomous systems is presented. The strategy is based on the probability of mission failure. There are two times at which mission failure probabilities must be calculated; these are described below, before the overall strategy is presented. The basic requirements for implementing such a strategy are then discussed.

3.1 *Mission Failure Probabilities*

There are two different probabilities that can be required as a mission progresses. These are:

- initial failure probabilities,
- updated failure probabilities.

The initial failure probabilities are calculated before a mission begins. For a single platform, these give a measure of that platform's failure probability in each of its phases and also of the total failure probability of the platform. The updated failure probabilities are calculated once the mission is in progress at the point that system functionality or environmental changes occur. They provide a measure of the failure probability given information about the system, such as the phases that have been completed or the known status of certain parts of the system. For multi-platform missions, the same initial and updated failure probabilities can be calculated for the single platforms performing the mission and also for the mission as a whole. Thus, the failure probability of the mission can be calculated for the mission in each of its constituent phases and also over its entire length.

3.2 *The Strategy*

Presented in this section is a decision making strategy for autonomous vehicles. The strategy involves calculating the probability of mission failure at required points within the mission (e.g. after faults are known to have occurred or after other environmental effects occur). If the probability of mission failure is unacceptably high then a mission reconfiguration occurs.

The algorithm for the decision making strategy is shown in Figure 2. There are two tools that must be used in order to facilitate the decision making: a prognostic tool and a diagnostic tool. The diagnostic tool must track the system (or system of systems in the case of multi-platform systems) and recognize a number of events, such as:

1. Any faults that occur within the system, such as faults occurring on individual platforms. Faults might be identified at component level but most likely at functional or subsystem level.

2. Environmental changes that affect the platforms, such as the weather or the presence of an enemy force.
3. Successful components of the mission. E.g. completed platform phases, mission tasks, or the known successful function of components, subsystems or systems up to certain points within the mission.

The job of the prognostics tool is to use all available mission information to provide measures of the likelihood of mission success. A reliability based technique is presented here to provide this measure. The tool calculates failure probabilities for both individual platforms and, in the case of multi-platform phased missions, for the mission as a whole. Failure probabilities are calculated for the separate phases of the mission and also for the entire duration of the mission.

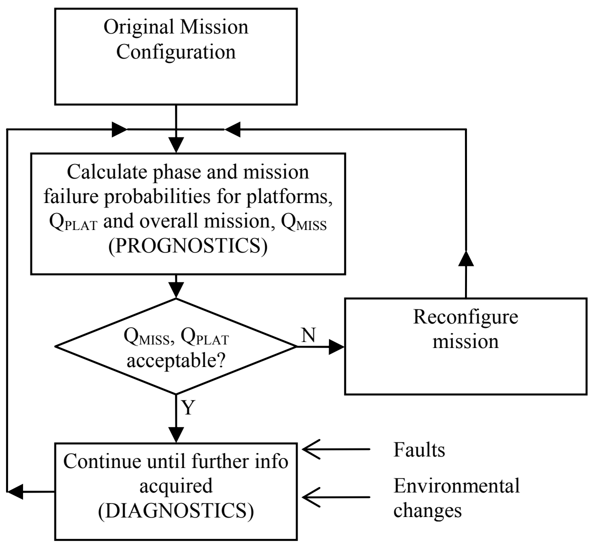

Figure 2 – The Decision Making Strategy

In order that the failure probabilities can be used as part of a decision making tool the acceptable mission failure probability is defined for each platform and for the mission as a whole. If the failure probabilities predicted by the prognostics tool exceed any of these values then there is a requirement to reconfigure the mission. This could mean that the platforms involved in the mission perform their mission tasks in a different way in order that the objective be achieved. Alternatively, the platforms could be reconfigured to work towards an alternative mission objective or the mission could be aborted altogether and the platforms return to base.

Thus, for a typical mission, a prognostics assessment would be carried out before the mission began. This would involve the calculation of the initial failure probabilities of the original mission configuration. If these failure probabilities were acceptably low then the mission would begin. Once the mission is under way a diagnostics tool would gather information about any changes to system variables. As new information is obtained, the prognostics tool would calculate updated failure probabilities which would again be checked against the acceptable values. If, at any time, the probability of failure reaches unacceptable levels then a mission reconfiguration would take place. This could be a reconfiguration of platforms or a reconfiguration of the tasks that they perform.

3.3 *Requirements*

Clearly, in most missions, the prediction time will be of critical importance. Having the ability to respond quickly to a changing environment or situation could, for example, give an edge to a force in a combat situation. It follows that, should autonomous systems be deployed, decisions should be well-informed and accurately made in the shortest possible time. This leads to two main requirements for the reliability-based prognostics tool. The failure probabilities calculated within it must be:

- Accurate.
- Quickly obtained.

If one considers the reliability tools available, Monte Carlo simulation [4] can be discounted due to the time taken to perform a sufficient number of simulations to obtain convergent results. Markov techniques [5] can also be discounted due to the complexity of the systems in question and the resultant state space explosion. Fault trees provide an excellent way of representing the failure logic associated with the platforms and their phases. However, when it comes to analysis, the kinetic tree theory [5] [6] will be unlikely to deliver the results required in the time available. Approximations would also need to be used, the accuracy of these approximations, particularly when NOT logic is incorporated in the system fault trees (i.e. the fault trees are non-coherent), may not be good enough. However, fault trees can be converted to binary decision diagrams (BDD), [7]. BDDs represent the failure logic of a system and can be analyzed to give exact values of system failure probability. The analysis is also very fast. Indeed, due to these properties, BDDs would appear to offer the greatest opportunity for performing the real-time analysis of the system failure probability that is required in the decision making tool.

Previous work has shown that it is possible to perform the analysis of single platform phased missions [3]. Methods of computing initial and updated failure probabilities are outlined in the next section for single platform phased missions. The extensions of these methods for multi-platform phased missions are then discussed.

4 *SINGLE-PLATFORM MISSION MODELING*

4.1 *Phased Mission Unreliability Quantification*

A critical part of the decision making strategy is for the prognostics tool to provide an accurate prediction of the probability of mission failure.

A method of calculating the mission failure probability is detailed in [3]. The method works by calculating the probability of failure, Q_i, in each of the mission phases, i, and then adding these to give the total mission failure probability, Q_{MISS}.

The failure logic for each of the platform phases is first represented in fault tree form. These must then be used to construct the failure logic representing mission failure in each of the phases. The general fault tree for this is shown in Figure 3. As can be seen from the diagram, in order for the failure conditions to be met in phase *i*, and for the mission to fail during phase *i*, the failure conditions must not have been met during any of the previous phases.

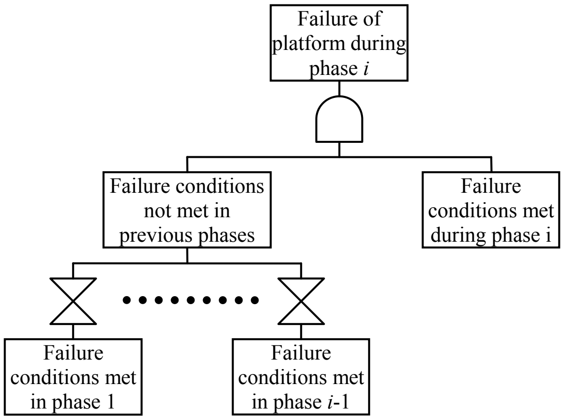

Figure 3 – Fault Tree for Mission Failure During Phase i

The analysis of this fault tree allows the probability of failure during mission phase *i*, Q_i, to be found. Thus the total mission failure probability is given by:

$$Q_{MISS} = \sum_{i=1}^{n} Q_i \qquad (1)$$

While updating the mission failure probability once the mission is under way [8], the fact that the previous phases of the mission have been successful is taken into account. Using Bayes' theorem to take into account that *k* phases have been successfully completed, phase *j* failure is given by:

$$Q_{j|\bar{k}} = \frac{Q_j}{1-\sum_{i=1}^{k} Q_i} \qquad (2)$$

Then Q_{MISS} is calculated by adding the phase failure probabilities of the remaining mission phases:

$$Q_{MISS} = \sum_{j=k+1}^{n} Q_{j|\bar{k}} \qquad (3)$$

Any additional information provided by the diagnostics system on the status of individual components, functions, subsystems or environmental effects must be included in the calculations for the updated phase failure probabilities. Equation (2) would again be used to incorporate this information.

Since efficiency and accuracy of the analysis is especially important in real-time modeling, phase fault trees are converted to BDDs offline prior to the mission. Using this method an accurate probability of the future mission failure can be provided to the decision making system in a short period of time.

4.2 *Platform and Mission Description*

Consider a simplified UAV that is to perform a phased mission. The mission objective is to travel from location A to location B and successfully land at location B. There are three mission phases to be performed, these being take-off, cruise and landing. All three phases must be successfully completed in order for the mission to be a success. Each phase may only begin after the successful completion of previous phases. It is important that the UAV makes clear, rational decisions and takes the action necessary to preserve safety and achieve the highest level of mission success. The following section gives an overview of the application of the decision making methodology.

4.3 *Interpretation of the Methodology*

Before the UAV can start its mission the initial phase failure probabilities, Q_i, are calculated by the prognostics tool using equation (1). Assume the values are as given in Table 1.

Q_1	Q_2	Q_3	Q_{MISS}
0.020	0.010	0.020	0.050

Table 1 – Initial Phase and Mission Failure Probabilities

If Q_{MISS} is within the accepted limits ($Q_{MISS}<Q_{max}$), the proposed mission can start. Otherwise, a new mission profile should be configured.

The updated mission failure probability will be calculated by the prognostics tool when some component failures occur and are reported by the diagnostic tool. For example, if during the take-off one of the functions of the flight control system is lost, using equation (2) gives the updated phase failure probabilities shown in Table 2.

Q_1	$Q_{2\|\bar{1}}$	$Q_{3\|\bar{1}}$	Q_{MISS}
0	0.300	0.200	0.500

Table 2 – Updated Phase and Mission Failure Probabilities

After each update Q_{MISS} is compared with Q_{max}. If Q_{MISS} increases too much, i.e. exceeds Q_{max}, a mission reconfiguration must take place. Alternative mission phases are incorporated into the mission, which result in a smaller mission failure probability or are perhaps necessary due to the circumstances observed. For example, the UAV might be redirected to another airfield, in order to land at the earliest opportunity, if the failure probability of the original mission is unacceptable. As the UAV continues its mission the decision making process will also continue.

5 *MULTI-PLATFORM MISSION MODELING*

5.1 *Phased Mission Unreliability Quantification*

The most important distinction between single platform phased missions and multi-platform phased missions when modeling reliability is that mission tasks and phases of different platforms are not necessarily sequential. It is important to identify the separate phases of the mission, given the platform phases and the way that the platforms must interact. This is done by identifying all distinct individual platform phase start and end points. This is best explained by means of an example covered in section 5.2.

Once the distinct mission phases have been identified the initial failure probabilities can be calculated for the single platforms using the techniques shown in the previous section. For single platforms operating within a multi-platform phased mission the failure probabilities are calculated in the mission phases rather than in the platform phases. A similar process is used when calculating the updated failure probabilities for the single platforms, with the probability of failure in the mission phases being calculated. The mission prognosis for the individual platforms would be compared against acceptable levels to determine the necessity of mission reconfiguration.

Calculating the initial probability of failure of the mission as a whole in the separate platform phases involves constructing the failure logic of the mission in the relevant phases. Given the assumption that the failure of any platform causes the mission to fail the logical expression for the probability of mission failure during a mission phase is constructed as follows. Account is taken of the fact that all platforms must have successfully completed the previous mission phases and any single platform failure during that mission phase causes mission failure. This logical expression is quantified to give the initial phase failure probability for the mission and these values are added to give the initial failure probability for the mission as a whole. Note that platform phase failure probabilities cannot simply be added to give mission phase failure probabilities because of dependencies that are likely to exist between platforms that are working in collaboration.

As with the single platforms, the updated failure probabilities are calculated using Bayes' theorem to take into account any information about the system, provided by a diagnostic tool. Phase failure probabilities would be calculated for each of the remaining phases and these added to give the total mission failure probability for the remainder of the mission.

5.2 *Platform and Mission Description*

Consider a UAV and an autonomous land vehicle (LV) that are required to work together to achieve a mission objective, as shown in Figure 4. As the mission is due to begin they are at separate locations. They must meet at a third location, A, before traveling together to a final destination, B, with the UAV performing reconnaissance ahead of the land vehicle.

Assume that the UAV must simply cruise to the meeting point and then fly a reconnaissance pattern ahead of the land vehicle, i.e. two distinct phases. Assume that the land vehicle must also perform two distinct phases, traveling to the meeting point and traveling to the final destination. There is a constraint on the platforms in that the land vehicle and the UAV are required to reach the meeting point simultaneously. Due to the distances to be traveled by the platforms and their speeds, the land vehicle must start its journey to A before the UAV. This leads to the mission representation shown in Figure 5, where it can be seen that, because the land vehicle takes longer to complete its first phase than the UAV, the UAV starts its first phase after the UAV. It can be seen that there are three distinct mission phases, which are:

1. The land vehicle travels to A,
2. The land vehicle travels to A AND the UAV travels to A,
3. The land vehicle travels from A to B AND the UAV travels from A to B while performing reconnaissance.

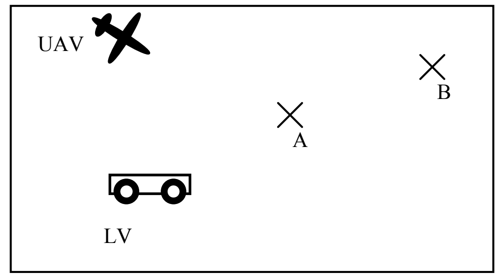

Figure 4 – Plan View of Mission Locations

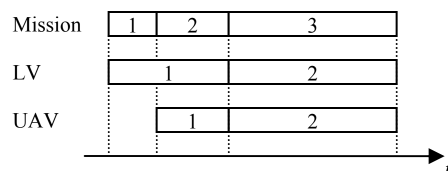

Figure 5 – Representation of Platform and Mission Phases

5.3 *Interpretation of the Methodology*

Before the mission begins the prognostics tool is used to calculate the initial phase failure probabilities for each platform and for the mission as a whole. These are added to give the total failure probability for the UAV, the LV and the mission as a whole over the entire mission duration. Assume that these values are as shown in Table 3.

Platform	Q_1	Q_2	Q_3	Q_{MISS}
LV	0.020	0.025	0.050	0.095
UAV	-	0.019	0.012	0.031
Mission	0.020	0.040	0.055	0.115

Table 3 – Initial Phase and Mission Failure Probabilities

Note that there is no failure probability associated with the UAV in mission phase 1, since the UAV only begins its own mission in mission phase 2. Let us assume that at this point the failure probabilities are acceptable and the mission begins. The LV begins its journey to point A and at some point soon after the UAV also starts to travel to A. They meet at A, at which point the UAV is due to perform reconnaissance ahead of the LV while it travels to point B. However, at this time, a fault has occurred in the communication system used by the two vehicles and is reported by the diagnostic tool. A new calculation of the mission failure probability is needed quickly, before mission phase 3 can begin. Updated phase failure probabilities are calculated and are as given in Table 4. Now the failure probabilities of each platform, and of the mission as a whole, have risen. In the case of the mission the

Platform	Q_1	Q_2	Q_3	Q_{MISS}
LV	0	0	0.150	0.150
UAV	-	0	0.075	0.075
Mission	0	0	0.210	0.210

Table 4 – Updated Phase and Mission Failure Probabilities

failure probability has risen to 21%. This is deemed unacceptably high. Thus, at this point a mission reconfiguration would take place. Options could be to take a different route to point B or perhaps abandon the mission and return to the starting points.

6 DISCUSSION

Clearly, throughout the course of any mission such as those described above for single platforms and multiple platforms, the speed at which decisions can be made about the future course of the mission is very important. Any way that the time taken to make decisions can be reduced will be beneficial. The calculations supporting the decision also need to be accurate.

It is proposed that BDDs should be used to manipulate and quantify the logical expressions required to calculate failure probabilities for the decision making strategy. However, due to the requirement to gain prognoses in as close to real-time as possible, it would be advantageous to store BDDs in an offline library ready for combination and quantification as necessary when the mission is in progress. In this way the construction of the main BDDs would be carried out before any mission is underway. Since BDD construction can be one of the more time-consuming parts of the process this saves on computation time during a mission. When a mission is underway an online combination of the previously constructed BDDs would provide the fastest possible results. Future research will look at applying the techniques outlined in this paper in the most efficient way possible.

REFERENCES

1. J. D. Esary, H. Ziehms, "Reliability Analysis of Phased Missions," *Reliability and Fault Tree Analysis*, 1975, pp 213-236.
2. G. Vyzaite, S. Dunnett, J. D. Andrews, "Cause Consequence Analysis of Non-Repairable Phased Missions," *Reliability Engineering and System Safety*, vol. 91, 2006, pp 398-406.
3. R. LaBand, J. D. Andrews, "Phased Mission Modelling Using Fault Tree Analysis," *Proceedings of the IMechE, Part E: Journal of Process Mechanical Engineering*, vol. 218, 2004, pp 83-91.
4. E. J. Henley, H. Kumamoto, *Reliability Engineering and Risk Assessment*, Prentice-Hall, 1981.
5. J. D. Andrews, T. R. Moss, *Reliability and Risk Assessment*, Longman Scientific & Technical, 2002
6. W. E. Vesely, *A Time Dependent Methodology for Fault Tree Evaluation,* Nuclear Design and Engineering, vol. 13, 1970, pp337-360.
7. R. M. Sinnamon, J. D. Andrews, "Improved Accuracy in Quantitative Fault Tree Analysis," *Quality and Reliability Engineering International*, vol. 13, 1997, pp 285-292.
8. J. D. Andrews, "Component Contributions to the Failure of Systems Undergoing Phased Missions," *Proceedings of the 17^{th} AR^2TS*, 2007, pp 155-168.

BIOGRAPHIES

John D. Andrews
Department of Aeronautical and Automotive Engineering
Loughborough University
Leicestershire, LE11 3TU, UK

e-mail: J.D.Andrews@lboro.ac.uk

John D. Andrews is Professor of Systems Reliability in the Department of Aeronautical and Automotive Engineering at Loughborough University, UK. The prime focus of his research has been on methods for predicting system reliability in terms of the component failure probabilities and a representation of the system structure. Much of this work has concentrated on the Fault Tree technique and the use of the Binary Decision Diagrams (BDDs) as an efficient and accurate solution method. He is the author of around 100 research papers on this topic and is joint author, along with Bob Moss, of a text book, Reliability and Risk Assessment, now in its second edition, published by ASME. John is a member of the Safety and Reliability group of the Institution of Mechanical Engineers and is the Founding Editor of the Journal of Risk and Reliability, which now forms part O of the IMechE Proceedings. He is also a member of the Editorial Boards for Reliability Engineering and System Safety, and Quality and Reliability Engineering International.

Darren R. Prescott
Department of Aeronautical and Automotive Engineering
Loughborough University
Leicestershire, LE11 3TU, UK

e-mail: D.R.Prescott@lboro.ac.uk

Darren R. Prescott is a Research Associate in the Risk and Reliability Group of the Department of Aeronautical and Automotive Engineering, Loughborough University. Darren gained a first class honours degree in mathematics from Loughborough and in 2003 was awarded a distinction in his MSc in Industrial Mathematical Modelling. Following this Darren completed a Doctorate project focussed on the systems reliability modelling of aircraft whose systems are subject to Time Limited Dispatch. His latest research interests focus on the use of systems reliability techniques in the decision making process for missions undertaken by multi-platform autonomous systems.

Rasa Remenyte-Prescott
Department of Aeronautical and Automotive Engineering
Loughborough University
Leicestershire, LE11 3TU, UK

e-mail: R.Remenyte-Prescott@lboro.ac.uk

Rasa Remenyte-Prescott is a Research Associate in the Risk and Reliability Group of the Department of Aeronautical and Automotive Engineering, Loughborough University. Rasa gained BSc and MSc degrees with honour in mathematics from Kaunas University of Technology, Lithuania. Following this Rasa undertook her Doctorate project at Loughborough University on systems reliability modelling of non-coherent systems using the Binary Decision Diagram technique. Rasa's current research interests involve the phased mission modelling of focus UAVs making predictions on mission success when failures occur or conditions change during the mission.