Using Health Information to Reconfigure Platform Operation, Adjust Mission Goals and Extend the Life of the System

James Kozlowski, Karl Reichard, Scott Laurin

Penn State Applied Research Laboratory
P. O. Box 30
State College, PA 16801-0030
814-863-3849
jdk173@psu.edu

Abstract

One of the primary advantages of having prognostic information about a platform is the ability to adapt operation and mission planning to compensate for the impending reduced capability. In addition, information about the operating ranges and mission requirements can be used to change the point in time the fault will occur. For example, a change in how the load on a power system is distributed may extend the point a fault will occur and therefore, provide additional mission time and/or life of the power system. Thus, there is a need to develop a unified approach to apply health information to operating conditions and mission planning to minimize the impact of the impending fault as well as use the operating limits and mission parameters to extend the life of the failing components or system to maximize the life of the platform and range of the mission. This paper analyzes and compares different approaches to applying health information to operation reconfiguration, mission planning and life extension. The weaknesses and strengths of these approaches are analyzed and a unified approach is proposed for classifying and analyzing health integration schemes for command and control structures. In addition, several health integration examples, from publicly available literature, are provided using the proposed classification and analysis approach to demonstrate its range and value.

I. Introduction

Among commercial, industrial, and government agencies, system health monitoring has become a key focus for reducing maintenance cost, improving safety, and increasing mission readiness. Due to improved technology and understanding of health monitoring benefits, diagnostic and prognostic information is becoming more readily available for many electrical and mechanical systems and integration of this information into operation and planning is starting to become a primary concern to command and control. Since the health of a platform may impact the objectives and safety of a mission, making the health information a part of the system dynamics and decision processing would ensure success of the mission and maximize the life of the system.

One of the fundamental concerns with integration of health information into the command and control structures is the form or level of the information. For example, some commercial health monitoring systems simply provide vibration signals and it is up to the user to interpret the signals and how to utilize that information. At the other extreme, some systems only provide the user with very high level information, such as the status of the system, whether the system is in good health or faulted. Thus, in one case the information is very raw and the user needs to process the data for the health information and in the other case, the information is very processed and the user has no details about how the health information was determined. Therefore, health information can be at various levels from raw data to processed messages; this will impact how the information can be integrated into the command and control loop. The level of information may be driven by a number of factors that include processing limitations, system complexity, methodology, etc. Hence, requiring a specific level the health information may exclude certain types of health monitoring systems and approaches. In addition, some command and control structures may only allow certain levels of health information to enter the loop.

Command and control structures are generally divided into one of three types: 1) Deliberative, which is based on logical planning and environment models, 2) Behavioral, which is based on reaction planning to sensed inputs, and 3) Hybrid or layered approaches that try to encapsulate features from both types. However, all of these structures can be captured in Boyd's OODA (Observe-Orient-Decide-Act) scheme of analysis of decision-making processes [2]. *Observe* functions refer to gathering, monitoring, and filtering data; *Orient* functions refer to deriving a list of options through analysis, trend prediction, interpretation, and integration; *Decide* functions refer to decision-making based on ranking available options; and *Act* functions refer to execution or authority to act on the chosen option. Thus, integration of health information into the different command and control structures can be characterized using the OODA description.

Currently, integration of health information into command and control is not very mature but has been recognized for many years as one of the key goals of using health information. Since health information and processing is also at an immature level (except possibly the diagnostic level of information), there has been some reluctance to develop integration methodologies because the health information may dictate how the integration can occur. However, there are some sources in the reviewed literature that have developed health integration strategies but these have generally been focused on specific applications. Thus, the rationale is to meet the direct needs of the specific application and adapt the health integration approach to other applications. This approach provides supporting evidence the integration technique can be successfully implemented, but does not guarantee the approach can be applied to other applications. Thus, if subsequent applications are similar to the original, extending this approach to other applications can likely be achieved. However, if there are applications in need of health information integration that are dissimilar to the original application, the approach may not be universal enough as a broad approach to health integration. In addition, the language used to describe the approach may be obscuring its applicability since it may have been more tailored to a specific application. Also, a field such as the paper industry may use very different descriptive language than the military autonomous vehicle community in terms of health monitoring and control. Thus, a "one-size-fits-all" approach to integration of health information into command and control structures and approaches that are too specific may exclude some applications. Therefore, one of the main goals of integrating health information is to establish a common framework to describe and analyze the inserted health information into the command and control loop.

The first section of this paper is a description of health information as it applies to machine diagnostics, advanced diagnostics, and prognostics. The second section describes common command and control structures and how they compare to each other. The next section covers the integration of health information into the different locations in the command and control loop and in support of this, a section providing examples of the integration process by using published sources that have developed approaches for feeding the health information into a command and control loop is provided. The last section of the paper reviews the approach in terms of potential application, where there needs to be additional development, and how this can impact reconfiguration of the platform operation, adjustment of mission goals and extension of the system life.

II. Health Monitoring

Since the term, *health monitoring* is a very broad term that covers a range of topics from diagnostics and prognostics to model-based and data-driven approaches, terminology is addressed in this section.

2.1 Types of Health Monitoring

Some of the fundamental questions that arise from available health information are the type of information that is provided as well as how the information was derived. Thus, these are the first two categories of health information that need to be classified in order to understand how this can be integrated in the command and control loop.

2.1.1 Diagnostics, Advanced Diagnostics, and Prognostics

Health information tends to be classified as either diagnostic, advanced diagnostic or prognostic information. A diagnostic assessment means that a fault or problem has already occurred and the information is after-the-fact. Advanced diagnostics refers to detection of a precursor of a failure or problem that has not yet occurred but will shortly. Some confusion about this term does arise from the prognostics perspective in that it is *predictive* of the impending fault. However, no real time frame can be attached to the information, only that indication of a particular failure mode is about to happen as opposed to a different failure mode or no signs of abnormal behavior. Thus, an advanced diagnostic assessment is a kind of "high-alert" but not ETA (estimated-time-arrival) of the fault. On the other hand, a prognostic assessment is an estimate of the time (or other lifetime metric such as cycles) until the fault will occur. Prognostic information can either be functional/operational or specific to a failure mode. For example, an operational prognostic assessment would predict at what point the system will be unable to perform a specific task or function and a failure mode prognostic assessment would predict when a specific failure mode has reached a defined level (e.g. gear tooth has cracked off). An operational prognostic assessment may have been provided because a specific failure mode cannot be clearly defined due to the system complexity or ambiguity of the fault and a failure mode type may have been provided since a operational level prognostic assessment may be compounded by other factors such as parallel system components and other influences outside the reach of the prognostic processing. Thus, how the health assessment is derived can influence where the information can be inserted into the command and control loop.

2.1.2 Data Driven, Model-Based and Other Processing Approaches

Processing of raw sensed data and available knowledge about the system being monitored will dictate how the health information can be defined: diagnostic but functional, prognostic and failure mode specific, etc. The processing itself may also be constrained by other factors such as system complexity, lack of comprehensive sensory inputs, and even limited processing power to extract the health information. Within the processing itself, there are generally two approaches to deriving the health information: data-driven and model-based approaches. Data-driven approaches usually do not require any knowledge of the system prior to operation and only require the system initially be in a good or normal state of operation so a baseline can be built. From there, the processing detects abnormalities or deviations from the baseline and makes adjustments based on user input. The advantage of this approach is the algorithms can be applied without a large (if any) knowledge about the system. In cases where the system is very complex, modeling may be difficult or not practical but a data-driven approach could still be applied. In some cases, up front training can be applied so some knowledge can be applied when a precursor or fault is encounter. However, a good prognostic assessment is where this type of approach begins to have difficulties. Since explicit knowledge of the system dynamics is not known, prediction of behavior changes can only be based on previously encountered scenarios. Thus, predicting how the system health will react to operational and environmental factors will be less accurate than an algorithm that models the system health. Many systems are beginning to utilize a combined approach that takes advantage of the best aspects of both data-driven and model-driven strategies.

Model-based approaches provide a means to "fast-forward" the model to predict the change in health and therefore, estimate the time to failure based on specific behavior of the system. Model-based approaches can range from physical dynamic models to functional or behavior-based models. In the first case, the models can be complete system models with health as either states or observable parameters or the models can be limited to the dynamics of the failure modes that are driven by the actual system. In the second case, if the physical dynamics are too difficult to characterize or too complete to process practically, a behavior or function model can be developed. However, unlike data-driven approaches knowledge of the system (even from a behavior or functional level) is used to model the health changes of the system. The primary hurdles to these types of approaches are system complexity and processing limitations. In addition, much more up-front training may be required to validate the accuracy of the model. However, a combination of model-based and data-driven approaches provides a means to adapt the processing as new information becomes available. Unfortunately, as with other model-based approaches this may drive up the processing requirements and the model complexity. Thus, whether the health information is only diagnostic or truly prognostic can depend heavily on the complexity, maturity, and practicality of the health processing. Furthermore, the health information itself can be provided at different levels to the user because of the above mentioned issues as well as needs of the command and control loop to integrate the health information.

2.2 Health Information Level

Another difficulty with health information is the level of information provided to the user. In other words, some health information can be as simple as raw vibration measurements and in other cases, the information may be highly processed as "system has a fault". The usefulness of the health information will also depend on the command and control loop. Some control loops may have a high level of complexity such that raw vibration data can be effectively integrated into the control so health can be accounted for. However, some control loops may be less complex and only high-level descriptions of health changes can be integrated. Thus, the level of health information provided to the user should fall into a class (or level) such that the command and control loop can use it to establish if and where health information can be inserted.

To better establish what level of health information is available to the command and control loop, six levels are being proposed here based on processing approaches and their associated output forms as well as standards used to classify types of data (e.g. OSA-CBM). The difference here is that no effort is being made to outline a standard form of the health information or data; just assign a level classification. A corresponding data format of health information can certainly be established (e.g. level 6 health information may be formatted as Decision Reasoning type data in OSA-CBM). However, the goal here is not to enforce any kind of data standards, but just to establish what level the health information is to better aid in determining the usefulness of the information to the command and control loop. Table 1 shows the proposed levels to classify health information output from diagnostic, advanced diagnostic, and prognostics algorithms and processing. The six different levels are divided into the three types of health information (diagnostic, advanced diagnostic, and prognostic) since some faults may only need to be diagnosed where other more critical faults may need to prognosticated. As an example of using this Health Information Level (HIL) classification, health information related to transmission faults is applied to the HIL classification as shown in Table 2 where a gear tooth fault is being monitored.

Table 1. Health Information Levels

Level	Diagnostics	Adv. Diagnostics	Prognostics
6	A general description indicating if the system or part is good or bad.	A general description notifying the user if there are significant signs that the system or part will fail within the mission or designated window.	A general description notifying the user when the system or part will fail under the constraints of the mission or designated conditions.
5	A functional or performance description of the failure that has occurred.	A functional or performance description notifying the user if there are significant signs that the system or part will fail within the mission or designated window.	A functional or performance description notifying the user when the system or part will fail under the constraints of the mission or designated conditions.
4	A physical or mechanism description of the failure that has occurred.	A physical or mechanism description notifying the user if there are significant signs that the system or part will fail within the mission or designated window.	A physical or mechanism description notifying the user when the system or part will fail under the constraints of the mission or designated conditions.
3	Parametric or model-based variables that indicate what the failure is.	Parametric or model-based variables that indicate significant signs that the system or part will fail within the mission or designated window.	Parametric or model-based variables that indicate when the system or part will fail under the constraints of the mission or designated conditions.
2	An extracted features or fused sets of signals that are associated with diagnosing the system or part.	An extracted features or fused sets of signals that are associated with indicators if the system or part will fail within the mission or designated window.	An extracted features or fused sets of signals that are associated with indicators of when the system or part will fail under the constraints of the mission or designated conditions.
1	Raw sensor signals associated with diagnosing the system or part.	Raw sensors signals associated with advanced warning of a failure of the system or part.	Raw sensor signals associated with detecting failure mechanisms and projecting the end of life of the system or part.

Table 2. HIL Classification of Transmission Fault Data

Level	Diagnostics	Adv. Diagnostics	Prognostics
6	"The transmission is broken."	"The transmission has signs of an imminent failure."	"The transmission will fail in 4 hours for this mission profile."
5	"Gears #2 & #8 are broken or damaged."	"Gears #2 & #8 are accumulating damage and will soon fail."	"Gears #2 & #8 will fail in 2.6 hrs given the current load and speed."
4	"Gears #2 & #8 have broken tooth meshing signatures."	"Gears #2 & #8 have gear meshing signatures that are growing fast as if a tooth is cracking."	"Gears #2 & #8 have meshing tooth cracking signatures and will crack in 2.6 hours."
3	"The 3rd harmonic of the mesh frequency for gears #2 & #8 has reached 4x the normal vibration level."	"The 3rd harmonic of the mesh frequency for gears #2 & #8 is growing at a rate of 5 g's every 1,000 cycles."	"The 3rd harmonic of the mesh frequency for gears #2 & #8 will reach 4x normal vibration at current growth in 230,000 cycles under the current torque load."
2	"The magnitude at 150 Hz of vibration signal #4 is above 30dB."	"The magnitude at 150 Hz of vibration signal #4 is changing at a rate of 2 dB every 2 minutes."	"The magnitude at 150 Hz of vibration signal #4 will reach 30 dB in 3 hrs at current rate of growth."
1	"Vibration sensor #4 = 40 dB rms"	"Vibration sensor #4 = 10 dB rms"	"Vibration sensor #4 = 25 dB rms"

At the lowest level, the health information is nothing more than raw sensor signals *related* to particular fault conditions and is up to the user to determine what this information mean to the health. At the next level, some processing (feature extraction, data fusion, etc.) related to particular fault conditions has been done to the health information. An example would be frequency spectra or order tracking data so the user could analyze and relate the features to the health of the system. At this level, no information about the specific system is accounted for (i.e. What does 30dB at 150 Hz mean? Gear mesh? Bearing cage?). Health information at the next level does include some information about the system being monitored. For example, in the case of the transmission, it knows that the vibration signature at 150 Hz is actually the 3rd harmonic of the mesh frequency from gears 2 and 8 and is reported to the user in this manner. However, in terms of interpretation of 30 dB for the 3rd harmonic, this is not indicated for this level of information. At Level 4, an association of the health information to a condition of the monitored system is provided (e.g. the transmission has a broken tooth on gear #2). At the next level up, health information would only indicate that the specific component or system cannot perform a given function or task. At the highest level, the health information only indicates if the system as a whole is able to perform a specific task or function. To clarify, at Level 5, health information provided would be "Gears 2 and 8 cannot handle loads greater than 550 in-lbs" and at Level 6, health information would have only indicated "Transmission cannot be engaged for reverse movement". There are some health monitoring systems that can provide multiple levels of information and the command and control loop could choose to use as many levels of information needed to integrate health changes.

In conclusion, health information can be diagnostic, advanced diagnostic, or prognostic and at various levels. The classification of the health information in this way provides the command and control loop a better means to determine if it can be integrated. From another point of view, this classification approach to health information provides a language for the command and control to define what type and level of health information would be required for effective integration. However, there are many types of command and control structures and integration of the health information is also dependent on these aspects as well.

III. Command and Control

3.1 Types of Command and Control Structures

There are many different names and terminologies for the different control architectures used in command and control, but most can be distilled down to fit into one of a few more general categories.

A *deliberative architecture* is based on planning and a world model. Data is collected, analyzed, and then used to plan future actions to achieve mission goals. This architecture is similar to the hierarchical architecture where the system is divided into different levels; higher levels keep track of overall mission goals and lower levels are responsible for solving problems that are encountered to enable mission success. There are several drawbacks to this type of system. The system is not flexible and any modifications or changes require a reconfiguration of the entire system. There is a long response time due to the communication delays between layers. These delays also inhibit the system in highly dynamic environments, preventing it from adapting to the changing surroundings.

A *reactive architecture* is based on the desired behavior of the system and missions. Each mission goal is assigned a behavior and the behavior reacts to the environment as it changes to achieve its goal. The overall behavior of the system is a result of the combination of behaviors for each goal. Due to the nature of this architecture, one behavior has the ability to affect others and possibly cause a deviation from mission goals, making this architecture potentially unpredictable. This is also known as a heterarchical system. It is very similar to hierarchical, but uses a parallel architecture that is more able to adapt to a dynamic environment.

A *hybrid architecture* takes the benefits of both the deliberative and reactive architectures while minimizing the detriments. It generally consists of three layers, a deliberative layer, a control execution layer, and a functional reactive layer. The hybrid allows the system to adapt to dynamic environments while still being reasonably predictable.

3.2 Boyd's OODA Loop

The OODA (Observe – Orient – Decide – Act) loop is a good example of a hybrid architecture. The system is broken into four levels, each responsible for different aspects of the system mission. Observe is responsible for gathering, monitoring, and filtering data. Data is collected from the environment and prepared to be analyzed. Orient derives a list of viable options via analysis, trend prediction, data interpretation and integration using the data provided from the observation level. Decide analyzes the available options based on ranking and determines the best course of action. Act executes or gives the authority to execute the chosen course of action. Figure 1 shows the connection of the OODA components.

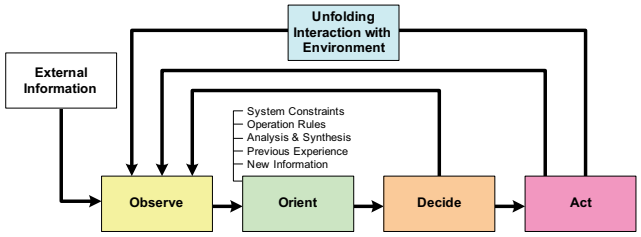

Figure 1. Boyd's OODA Loop [2], [3]

IV. Insertion of Health Information into Command and Control

Certainly, there are a number of different command and control structures and using the OODA loop description allows us to characterize most command and control structure using the same terminology. Using this approach, insertion of health information into the command and control loop can be described and analyzed more universally. There are potentially four entry points for the health information into the loop: the Observe, Orient, Decide, and Act segments. There can be scenarios where health information enters more than one of the four points in the loop and some health information may not be defined in a way that makes it clear that it is able to be inserted into any of the entry points in the loop. There will always be some exception to the rule, but as long as a majority of information can be inserted in one of the four entry points and exceptions can be accounted for, determination and analysis of health information inserted into the control loop can be achieved.

4.1 How Health Information Enters the OODA Loop

In order for health information to effectively improve command and control, the point in which the information enters the loop needs to be considered. In addition to the needs of the control loop, other factors such as type and level of available health information can impact the effect health information will have on mission goals and overall life of the system. Thus, examination of health information inserted into the four possible points of the OODA loop is addressed as well as insertion at multiple points.

Observe

The Observe element of the control loop represents the point where raw data is collected and filtered and certainly health information can be treated as raw data and inserted into the control loop as such. This means the health information is treated as an observed state or behavior of the system. Thus, the Orient block must be modified to be aware and accept health information coming from the Observe block.

Orient

The Orient element of the control loop represents the point where command and control options are derived based on the information received from the Observe block. So, health information that comes into the loop at this point has to appear as a command or control option. Thus, the health monitoring processing must be able to produce command and control options that are inserted into the loop at this point.

Decide

The Decide element of the control loop represents the point where a command is selected from the derived list of options from the Orient block and health information entering the loop at his point has to act as a steering or weighting of the possible command and control options. At this stage, the health information does not generate any new command and control functions, but instead contributes to which control actions are selected based on changes in the system health.

Act

The Act element of the control loop represents the point where the selected command option is executed or authorized to be acted on. Health information entering the loop at this point acts as an override or limiting function on the executed command. Thus, given the health condition of the system, a command can be cancelled to prevent further damage or limited to minimize the growing damage.

Overall, when health information enters the loop at the Observe point, the control options are not generated from the health data but instead contribute to the inputs used to generate control options. Health data that enters the loop at the Orient level must enter as a control option to contribute to the set of control options generate by the existing control loop. Thus, raw data (HIL 1) to overall system health (HIL 6) cannot directly enter the loop without being translated to some kind of control option that can be used for the decision processing. At the Decide level of the control loop, health information does not input any control options, but instead contributes to the decision processing by influencing the selection of the control or command based on changes in the health. So, whether the health data is raw (HIL 1) or overall system health (HIL 6), the information must be translated to a weighting factor or logic implication that contributes to the decision processing that selects the control option. However, if the control loop processes all the control-related information and the health information is inserted after this, it acts like an override or command limiter. Therefore, the health information must be translated into conditional statements that prevent or limit particular command and control options that were selected.

4.2 The Implications of the Different Insertion Points

Of the four possible entry points, only the *Observe* block allows direct insertion but this also puts the highest burden on the control loop to include the health information in control processing. For health information to enter the *Orient* block, the data has to be in the form of command or control options. Thus, the control loop is not required to fuse the information with other control-related information to generate control options, but the health processing must have some knowledge of the control mechanics to convert the health information into control options. However, for the *Decide* block the health information does not have to be converted into command or control options but still has to be converted into some kind of weighting factor or addition to the decision logic of the control loop. Thus, the health information processing needs to have knowledge of how the loop makes decisions from the generated control options. At the *Act* point in the loop, the health information processing does not have to have knowledge of the control option generation or selection processing, but does need to have knowledge of the impact of the control option executions. Thus, the health information has to be converted into a processing block that limits or cancels control actions based on the health conditions. Thus, depending on the entry point, conversion of the health data to command and control related information may be necessary as shown in Figure 2.

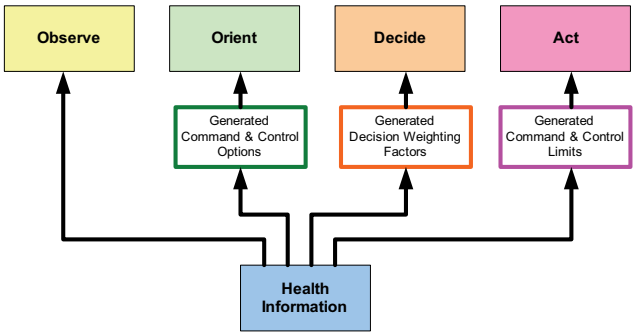

Figure 2. Insertion of Health Information into the OODA Loop

Certainly, a control loop that can accept direct health information would be easier on the health monitoring end, but now the control loop needs to know how to handle that information. If a new type or level of health information becomes available, the control loop will have to be modified to handle this. However, if the health information is processed to produce command and control information less adjustment to the control loop will be needed. Thus, generated control options can be the result of different types and levels of health information and if the health monitoring system is taken offline the control actions can still precede. This is also true for decision and action level input from the health monitoring system. For some types of systems, it may be difficult to form new control actions based on the health information, but may be easier to impose weighting on the different control options. Also, there may be control schemes where the system complexity makes it even difficult to establish weighting of the control options and at best the health can limit or prevent execution of control actions based on health conditions. At the other end of the spectrum, there may be control schemes where having the health information enter the loop at multiple points adds some required redundancy or entry at multiple points may be necessary based on availability of different types and levels of health information. Certainly, insertion may even be affected by the paradigms of a particular industry or even practical issues that limit where information can be inserted.

4.3 Constraints and Ambiguous Information

In some control applications where it is desired to include health information to improve performance and system life, the health information may seem too ambiguous such that the insertion point may not be clear. In other words, a given piece of health information may not appear to aid in the construction of control options; health information may not be complete enough to be converted into a control option; the weighting of the health information to different control options does not appear to influence the ultimate control decisions, or the connection to control actions is not apparent. In addition, constraints within the health information may restrict the possible insertion points. For example, insertion at the Observe level may require a specific rate of updating for the health information to be included into the control option building processing and the health information available may not be able to meet that constraint. Thus, this will limit the entry point of the health information in the control loop. Therefore, the best use of the health information may be constrained due to compatibility issues such as update rates and also, in terms of logical and useful connection to the control related information. The end result of determining the best use of health information may be to not use the information at all unless changes, additions, or different health information can be made.

4.4 Mission Planning, Reconfiguration, and Life Extension

Ultimately, the goals of using the health information are to make changes in the mission planning, reconfigure the system, and/or extend the life of the system. In terms of mission planning, health information reveals limitations in possible tasks as well as potential plans that would result in system faults. Thus, the most direct connection to this would be integration of health information at the Decide

level since control options (or mission plans) are additionally weighted by the health information. At the Observe level, the health information is included in the decision building and therefore more indirect but expected. For insertion at the Orient level, health information would be inserting new planning options. Thus, at this level the control scheme would have to be able accept new options as opposed to the Decide level where all the options are known and the health information is just aiding in the decision process. At the Act level, the health information acts to override or impose limits on tasks generated from the mission planning. Thus, for deliberative type control architectures insertion of the health information at the Observe and Decide level wound have the most impact and for the reactive type control architectures, insertion of the health information at the Orient and Act levels would have the most impact.

System reconfiguration as a result of changes in the health of the system provides the system the opportunity avoid faults, unsafe situations, and performance degradation. Health information at the Observe level is fused with other control information and therefore provides opportunity to reconfigure the system through the control option building processes. Also, at the Orient level the health information can be used to reconfigure the system by providing control options that reconfigure the system. The difference at this level is the control options are additional options input to the loop along with the normal control options generated by the loop; Observe level health information modifies the normal control options generated by the loop. At the Decide level, the reconfiguration options would have to already be included in the possible set of options and the health information would be used to select the options that reconfigure the system. At the Act level, the health information can only cancel or limit control actions and therefore, system reconfiguration is unlikely at this point in the control loop.

In addition to fulfilling the planned mission and avoiding faults and maintaining safety, extending the life of the system is another potential use of health information. Thus, the health information prevents overstressing components in the system. However, at some point this may result in making the system unable to complete a planned mission. Thus, the decision to abort a mission to extend the life of the system needs to be established by the operator. Certainly at the Observe level, command options can be adjusted by the health information to prevent overstressing the system. At the Orient level, the command options would have to be augmented with command options generated by health information that are focused on extending system life. This may result in command options that prevent completion of the desired plan but will aid in the extension of the life of the system. Health information inserted at the Decide level would steer the decision processing to commands that help extended the life of the system. At the Act level, the health information would be used to prevent execution of commands that overstress system that would result in shortening the life of the system. In conclusion, the system operator must prioritize mission planning, reconfiguration, and life extension of the system when changes in the health occur to establish the best insertion points for the health information.

V. Examples of Integration at Different Levels

To show the value of using this approach to analyze and determine the possible insertion points for the health information, several examples are provided using existing methods that have been proposed and published in the available literature on the subject of integration of health information into command and control.

A Power System Example Using Orient and Decide Level Integration

Consider the diagnostic integration solution proposed and evaluated in reference [5]. At the core of this approach "reasoning capability that could be included in the system to accomplish fault detection, isolation, reconfiguration and recovery" is emphasized as the key aspect. Thus, not only is a health monitoring approach developed but a means to integrate the information into the control of the system. So, in addition to sensing and algorithm processing, functions of the approach to "declare not only a viable, but optimum reconfiguration action, and *commands* the system to reconfigure to the new state" is developed. Thus, the approach is inserting health information at the Orient level since the approach converts the health information to command options. In addition, there is insertion at the Decide level since the system is required to follow the reconfiguration commands. As opposed to a system that only provides command options and allows the normal control processing to select or ignore the new commands generated from the health information. Although this approach can be applied to other types of applications, the health monitoring system must have some knowledge of the system in order to generate new reconfigurations of the system. As previously stated, some applications may have systems where the system is too complex or ambiguous that new reconfigurations or commands may not be possible. For some applications, the health information may have to be integrated at a lower level (Observe) or only at the execution level (Act) to prevent control actions that accelerate the fault damage or prevent effective response of the system.

A Propulsion System Example Using Observe Level Integration

For reference [7], propulsion systems were considered for development of health management systems that could "autonomously adapt to fault and/or contingency conditions with the goal of still achieving mission objective." Figure 3 show the integration of the health information into the control structure using a proposed approach for meeting mission goals under changing health conditions.

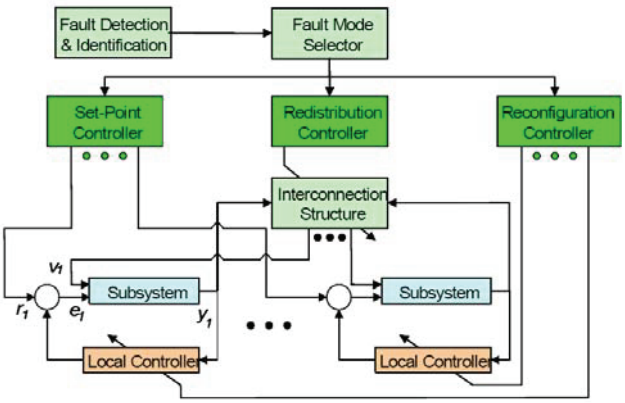

Figure 3. Integration of Health Information into a Propulsion System [7]

For this implementation of health information integration, the command decisions are readjusted based on health information. Thus, the health monitoring system does not generate new commands but instead, adapts or modifies existing command and control options. So, this integration approach inserts health information at the Observe level since no new command options are generated and the existing control is adjusted at a low level. For this type of approach, the existing control loop has to be able to accept health information as a form of sensor input and because of this the health information will be constrained to fit with the control system processing capability (data rates, number of inputs, etc.). New or additional health information would require the control loop to be modified. However, when good dynamic models exist for the system, such as the propulsion system in this reference, more granularity of influence of the health information can be achieved as opposed to an Act level integration that may only be able to cancel a command or hard limit a control signal.

A Unmanned Aerial Vehicle Example Using Act Level Integration

In reference [10], the problem of incorporating health information into a multiple autonomous vehicle application where the vehicles all work together to complete one mission was considered (Figure 4). In this case, health information about each vehicle has to be considered as a whole to the mission. Thus, some actions of an individual vehicle may be limited even though that particular vehicle does not have any health issues. In other words, the health information is entering that individual vehicle at the Act level of the control loop. The vehicle itself continues to sense the environment and accept plans from the host planner and make decisions but can be overridden or limited by the health information. This is done in this manner because the individual health of one of the vehicles in not as critical as the health of the entire unit of vehicles. Therefore, planned actions will be adjusted to accomplish the mission as opposed to using the health information to continue to execute given plans by the host planner

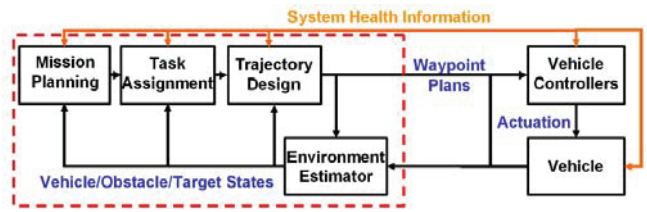

Figure 4. Health management for multi-vehicle mission systems [10]

Hence, with three examples of approaches to integrate health information into the command and control of the vehicle, three different approaches were developed. However, using the OODA view-point of integration of health information into the command and control structure, each could be classified as being an application of this integration process. From an application point of view, it is reasonable to understand how and why the approaches were derived. Unfortunately, the application of the first example may have found the approach developed from the third example unsuitable and vise-versa. Thus, having the unified approach to analyzing the different types of integration approaches allows for better comparison and analysis. So, other applications in need of health information integration can better determine which approach (or combination of approaches) will best meet the needs of the applications.

VI. Conclusions and Recommendations

At a very basic level, the process outlined in this paper provides a means to classify different types of approaches for insertion of health information into the command and control loop of systems. For the most part, the application dictates the how the health information can be inserted into the control loop. Other issues also arise such as limited access to different parts of the control loop, compatibility of data rates and types, highly complex and ambiguous systems, and industry paradigms. Thus, the OODA

description approach provides a common language for the different approaches, making analysis and comparison more accessible. In addition, this approach provides a starting point to selecting and/or designing the health information integration scheme for a selected application.

There will certainly be some applications where it will be unclear where the health information should be integrated and gaps such as this will provide insight into the next step towards improving this approach for classifying and analyzing health information integration into command and control structures. Nevertheless, this approach can aid in establishing if the selected approach can meet the goals of system reconfiguration, mission planning, and/or life extension of the system. These objectives also contribute to the type and level of health information and insertion points into the command and control loop required and the OODA description approach can reveal where this is best achieved. The limitations of this approach are ones of details of the specific application; this approach is a broad assessment of the required health integration approach. However, this broad view provides a large scale assessment before the specifics related to a particular application are addressed. A recommended next step for this approach is further evaluation of proposed health integration schemes and experimentation and analysis of health integration at the various insertion points to identify where further development is needed.

References

[1] R. C. Luo, C. C. Yih, and K. L. Su, "Multisensor Fusion and Integration: Approaches, Applications, and Future Research Directions," *IEEE Sensors Journal*, vol. 2, no. 2, pages 107-119, 2002.

[2] H. Hu, R. W. Proud, and J. J. Hart, "Spacecraft Mission Assessment and Replanning Tool (SMART): A real-time, intelligent autonomous flight management (AFM) system for increased safety and performance of human spaceflight vehicles," *AIAA Aerospace Science Meeting and Exhibit*, pages 11828-11838, 2004.

[3] S. Adolph, "What Lessons Can the Agile Community Learn from a Maverick Fighter Pilot?" *Proceedings of the AGILE Conference*, pages 94-99, 2006.

[4] R. N. Davis, M. E. Polites, L. C. Trevino, "Autonomous component health management with failed component detection, identification, and avoidance," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 219, no. 6, pages 483-495, 2005.

[5] M. Nolan and J. P. Giordano, "Use of adaptive model-based reasoning for embedded diagnostics and redundancy management for fault tolerant systems," *Proceedings from the IEEE Systems Readiness Technology Conference*, pages 455-466, 1997.

[6] M. Schwabacher, J. Samuels, and L. Brownston, "The NASA integrated vehicle health management technology experiment for X-37," *Proceedings of SPIE – The International Society for Optical Engineering*, v 4733, pages 49-60, 2002.

[7] L. Tang, A. Saxena, M. E. Orchard, G. J. Kacprzynski, G. Vachtsevanos, A. Patterson-Hine, Ann, "Simulation-based Design and Validation of Automated Contingency Management for Propulsion Systems," *Proceedings of the IEEE Aerospace Conference*, pages 1 – 11, March 2007.

[8] P. F. Lichtenwalner, E. V. White, and E. W. Baumann, "Information processing for aerospace structural health monitoring," *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 3326, pages 406-417, 1998.

[9] D. E. Paris, L. C. Trevino, M. D. Watson, "A framework for integration of IVHM technologies for intelligent integration for vehicle management," *Proceedings of the IEEE Aerospace Conference*, pages 559-569, 2005.

[10] M. Valenti, B. Bethke, J. P. How, D. Pucci de Farias, and J. Vian, "Embedding Health Management into Mission Tasking for UAV Teams," *Proceeding of the 2007 American Control Conference*, pages 5777-5783, July 2007.