Prognostics-enhanced Automated Contingency Management for Advanced Autonomous Systems

Liang Tang, *Member, IEEE,* Gregory J. Kacprzynski, Kai Goebel, Abhinav Saxena, *Member, IEEE*, Bhaskar Saha, *Member, IEEE,* and George Vachtsevanos, *Senior Member, IEEE*

***Abstract*—This paper introduces a novel Prognostics-enhanced Automated Contingency Management (or ACM+P) paradigm based on both current health state (diagnosis) and future health state estimates (prognosis) for advanced autonomous systems. Including prognostics in ACM system allows not only fault accommodation, but also fault mitigation via proper control actions based on short term prognosis, and moreover, the establishment of a long term operational plan that optimizes the utility of the entire system based on long term prognostics. Technical challenges are identified and addressed by a hierarchical ACM+P architecture that allows fault accommodation and mitigation at various levels in the system ranging from component level control reconfiguration, system level control reconfiguration, to high level mission re-planning and resource redistribution. The ACM+P paradigm was developed and evaluated in a high fidelity Unmanned Aerial Vehicle (UAV) simulation environment with flight-proven baseline flight controller and simulated diagnostics and prognostics of flight control actuators. Simulation results are presented. The ACM+P concept, architecture and the generic methodologies presented in this paper are applicable to many advanced autonomous systems such as deep space probes, unmanned autonomous vehicles, and military and commercial aircrafts.**

***Index Terms*—Automated contingency management, Prognostics and Health Management, autonomous system, control reconfiguration**

I. Introduction

Growing demand for improving the reliability and survivability of safety-critical aerospace systems has led to the development of prognostics and health management (PHM) and fault-tolerant control (FTC) systems. Traditionally, FTCS are classified into two types: passive and active. Passive FTC systems are designed to make the closed loop system robust against system uncertainties and some restrictive faults. For this reason passive FTC systems have limited fault-tolerant capability. In contrast Active FTC systems react to system component failures actively by reconfiguring control actions so that stability and acceptable performance of the entire system can be maintained [1]. Active FTC techniques that are capable of detecting the occurrence of faults while still retaining acceptable performance in the presence of faults are being developed for both manned and unmanned air vehicles [2]-[5].

The emergence and successful applications of PHM technology over the last decade, especially the development of on-line prognosis techniques, gave rise to a new category of FTC system, namely proactive FTC system. The two primary goals of proactive FTC are damage avoidance and ensuring primary mission success. Given accurate online prognostic information, proactive FTC system manages the accumulation of further damage through control actions, for example, by either reconfiguring the control settings at the controls level, or by re-planning mission profile through trading off secondary mission goals at the mission management level. We have introduced the Automated Contingency Management (ACM) concept in our previous work ([6],[7]) that extends traditional FTC at low level control to fault management at higher levels.

In this context, the term Automated Contingency Management (ACM) has been introduced to describe intelligent systems capable of mission re-planning and control reconfiguration based on health diagnostic information. Most fault detection and fault accommodating control techniques found in literature are based on diagnostics. They react to and compensate for faults and performance degradation after they are detected, but they are by definition a reactive paradigm. Well-designed reactive ACM systems may survive a severe failure, but may have missed the opportunities to mitigate or postpone the occurrence of the failure, or the opportunities to minimize the affect of the failure by mission re-planning.

This paper introduces a proactive prognostics-enhanced ACM (ACM+P) paradigm based on both current heath state (diagnosis) and future health state estimates (prognosis). Including prognostics in the control/planning loop poses several challenges. First, uncertainties associated with future

Manuscript received May 18, 2008. This work was supported in part by the NASA under STTR Contract NNX08CD31P.

Liang Tang and Gregory J. Kacprzynski are with Impact Technologies, LLC, Rochester, NY 14623 USA (Phone: 585-424-1990; fax: 585-424-1177; e-mail: Liang.Tang@impact-tek.com).

Kai Goebel is with NASA Ames Research Center, Moffett Field, CA 94035 USA.

Abhinav Saxena is with RIACS, NASA ARC, Moffett Field, CA 94025

Bhaskar Saha is with MCT, NASA ARC, Moffett Field, CA 94035

George Vachtsevanos is a Professor Emeritus at Georgia Institute of Technology, Atlanta, GA 30332 USA.

state prediction will, in general, increase as the prediction horizon increases so adaptive prognosis routines and uncertainty management capabilities are critical. Secondly, the larger time horizon afforded by prognosis requires ACM+P system to be split into real-time "reactive/proactive" components at lower levels and non-real time "planning" components at higher level. Thirdly, the performance of ACM+P system relies on the accuracy of the prognostic routines, and careful false alarm mitigation is needed.

The rest of this paper is divided into two main parts. Part One (Section II) establishes the general concept, architecture and presents the generic methodologies which are applicable to a broad range of autonomous systems. Part Two (Section III) is a case study, where the techniques described in Section II are applied to an UAV platform and simulation results are presented. The paper concludes with remarks on the main contributions of the presented work, technical challenges and future developments.

II. ACM+P System: Concept, Architecture & Generic Methodologies

A. ACM+P System Overview

Conceptually, ACM+P system is a system that is designed to provide the ability to proactively and autonomously adapt to current and future fault and/or contingency conditions while either achieving all or an acceptable subset of the mission objectives. An ACM+P system is different from a fault tolerant control system mainly in two aspects: 1) it consists of not only low level control reconfiguration, but also high level (mission) planning and optimization; 2) it uses not only diagnostic information, but also prognosis.

Although long term prognosis (with a remaining life estimate of several months or years in the future in our context) are mainly used to plan the system's maintenance schedule as what prognostics is intended for in the current state of the art, it also provides valuable information that can be utilized to establish realistic long term resource management and mission planning. The high level planner in ACM+P system enables the ACM routine to not only address immediate fault conditions but also establish a long term operational plan that will optimize the utility of the entire system. In addition, prognostics can be considered in control actions to manage the life of the component if performance requirements can be relaxed. Short term prognosis (with a remaining useful life estimate in terms of minutes or hours in the future) provides important information which can be utilized by ACM+P system to accommodate faults or mitigate failures.

A typical ACM+P implementation usually utilizes a hierarchical architecture as shown in Fig. 1. The PHM and situation awareness modules provide fault diagnostics, prognostics and contingency information to the ACM+P system, which in turn, identifies and executes the optimal fault accommodation and/or mitigation strategies. While PHM is a precondition for ACM+P, it is not considered a part of the core ACM+P system.

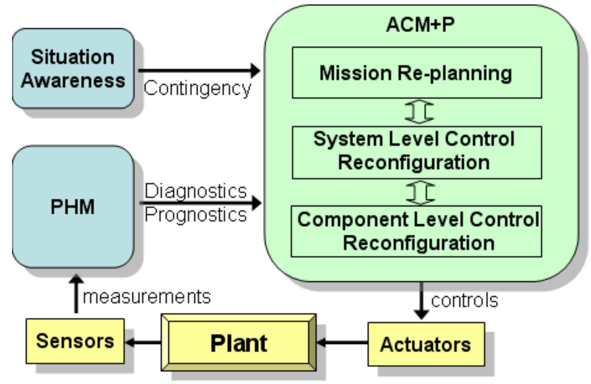

Fig. 1. Conceptual ACM+P System Hierarchy

Some important common features of the ACM+P system include:

(1) Hierarchical architecture;

It is important to realize that a component fault can often be accommodated at different levels in the ACM+P hierarchy and the decision should be made based on performance requirement and safety consideration. For example, if one of the two engines on a fixed-wing UAV is experiencing a severe degradation, the thrust difference may generate an unwanted yaw movement. This fault can be accommodated at the lowest (component) level by adjusting the fan speed (N1) set point value in the engine controller or by activating more sophisticated thrust control strategy [21]. It can also be accommodated by the trajectory following auto-pilot by rudder control.

(2) Use of redundancy and trade-off;

Traditionally, it has been possible to accommodate faults only in a system with redundancy, either physical redundancy or analytical redundancy. More advanced systems may include online healing concepts, including self-healing. When system performance can not be totally recovered by the fault accommodation strategies, trade-off of mission objectives has to be made to secure the most important tasks.

(3) Online optimization;

If ACM+P system is to be applied to a complicated system-of-systems platform such as the Crew Exploration Vehicle (CEV), it is often unavoidable to phrase the solution search as a dynamic optimization problem. This optimization problem may need to be solved online to arrive at the optimal contingency management strategies constrained by the available performance and resources to meet multiple (sometimes conflicting) mission objectives. It is important to realize that the optimization problems at different levels in an ACM+P system have different time horizon and real-time execution considerations.

(4) Uncertainty management and false alarm mitigation;

False alarm mitigation has been investigated in the context of PHM and FTC systems [9]. The use of prognostic information in the ACM+P system brings new challenges to both uncertainty management and false alarm mitigation. Since prognosis projects the current system condition in time using a prognostic model in the absence of future measurements, it necessarily entails large-grain uncertainty. This uncertainty has to be handled both in high level mission re-planning and middle/low level control reconfiguration modules.

In the following sections, some generic ACM+P methodologies are presented. Applications of these techniques are illustration in Section III with an UAV case study.

B. Control Reconfiguration in ACM+P System

As mentioned before, the proposed approach relies not only on current system performance/fault information (diagnosis) but also incorporates the projected future condition of the system (prognosis). By incorporating the likely future condition of the system into the ACM routine, it is possible to mitigate or at least prepare for the occurrence of a possible severe failure, so that the system will fail "gracefully" if the failure can not be mitigated with the control authorities available.

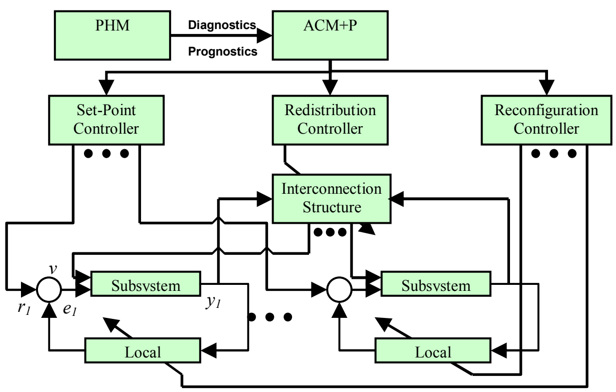

Fig. 2. Control Reconfiguration in ACM+P System

Conceptually, the middle and low level ACM strategies can be implemented within a generic architecture shown in Fig. 2. In general, the control reconfiguration module conducts three tasks to optimize the system's response to a fault (either current or future). First, the redistribution controller restructures system inter-connections to optimize the use of control authority. Then, based on the output of the redistribution controller, the set-point controller adjusts the set-points of the low-level controllers. Finally, the low-level gain controller adjusts the gains of the low-level controllers as necessary [10].

Implementing this architecture on a specific system requires the development of algorithms to perform each of its separate functions. That is where some of the real technical challenges are found. To give a flavor of these challenges, we provide a concrete example that describes how these control reconfiguration strategies can be implemented in a real system, in Section III. In particular, switching to main rotor RPM control is an example of "redistribution controller"; controlling the set point of main rotor RPM in the *RPM controller* is an example of "Set-Point Controller"; and changing the actuator constraints in the *prognostics-enhanced failure prevention controller* is an example of "Reconfiguration Controller". Similar examples for engine FTC and aircraft ACM (without the prognostic element) using this architecture can be found in our previous work ([6]).

C. Optimization-based ACM+P System

Most fielded FTC systems rely on heuristic information about a reduced set of severe, frequent and testable fault modes, a reasonable number of active controllers and a mapping between the fault modes and the control reconfiguration routines. Such a strategy has the ability to adaptively switch from one controller to another, if control limits are reached and by this switching action critical mission objectives can be realized. In this paper, we present a new approach in which an optimization problem is dynamically formulated and solved on-line to solve the optimal contingency strategy constrained by the available performance and resource. A typical cost model is expressed in terms of such elements as on-time execution of critical components, time to complete the mission, fuel consumption etc. Costs are computed online and may change dynamically.

Analytically, the objective of the ACM+P system is to optimize the utility of the vehicle with impaired capability to accomplish an assigned mission. The ACM+P system can be formulated as an optimization problem at two levels,

Mission planning (high) level:

$$J(M) = \max_{M} U(P_e, P_r, M, M_{com}) \tag{1}$$

Control reconfiguration (middle and low) level:

$$J(R) = \max_{R} P_e(F_m, P_r, R, M) \tag{2}$$

At the high level, mission adaptation and resource redistribution allows the control architecture to pursue relaxed mission objectives (M instead of original mission M_{com}) in order to achieve greater vehicle usefulness (U) based on closed loop performance (P_e) and prognostic information (P_r). At the lower level, the objective is to optimize vehicle performance P_e while satisfying the mission constraints, through restructuring and reconfiguration, R under the current fault condition F_m. This formulation is based on the work in [11] with P_r added to include prognostics in the optimization problem. Practically, the above optimization problems have to be solved while adhering to various constraints including system dynamics and resource limitations.

To facilitate the formulation of the optimization problem, an ACM guarded system can be represented by a Finite State Machine (FSM) as shown in Figure 3. There can be multiple states in each of the three state-spaces, but the general nature of transitions between different states can be described by five types of transitions as depicted. With the modeling paradigm described above, the ACM algorithm can be formulated as a constrained optimization problem as stated: given the current states of the system, and subject to predefined system constraints, find the optimal action series that will bring the system to the desired states with minimal cost.

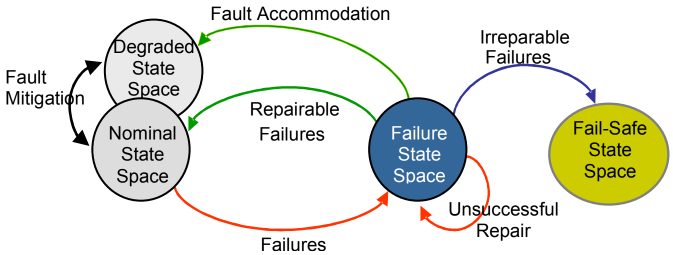

Fig. 3. ACM Modeling

D. *Mission Re-planning with Prognostics Consideration*

Prognostic information concerning the changes in the health condition in the future will enable the ACM algorithm to derive long term strategies and thereby minimize the overall effect of the fault. To illustrate how prognostics can be integrated into the optimization problem, a typical mission planning problem is used here as an example.

Suppose a system (such as an UAV) is assigned k tasks: $S=\{s_1,s_2,...s_k\}$ (e.g. a set of points of interest to visit). The importance of task s_i is denoted c_i and the expected time and resource (e.g. fuel) required to accomplish the task is t_i and r_i respectively. The mission planning routine will find the sequence of tasks that maximizes the mission success criterion subject to time and resource constraints. For simplicity, it is assumed that the tasks are independent of one another. The optimization problem can therefore be formulated as,

$$\begin{aligned} &\max_P C\cdot P \\ &s.t. \quad T\cdot P\le T_{\lim} \\ &\qquad\;\; R\cdot P\le R_{\lim} \end{aligned} \tag{3}$$

where, $C=[c_1,c_2,...,c_k], T=[t_1,t_2,...,t_k], R=[r_1,r_2,...,r_k]$, $T_{\lim}$ and $R_{\lim}$ are the maximum time and resource allowed for all tasks; $P=[p_1,p_2,...,p_k]^T$, $p_i\in\{0,1\}$ is the decision variable. When $p_i=1$, task s_i is chosen. With this problem formulation, T and R are potentially affected by fault evolution which can be estimated using prognosis information. $T_{\lim}$ and $R_{\lim}$ may also need to be updated to allow enough time and resource for successive missions.

Furthermore, to capture the uncertainties inherently present in the prognostics routine, the high level optimization can be formulated as a stochastic programming problem. The deterministic optimization approach without consideration of prognostic information may in fact not be optimal over a longer period of time.

E. *Control Reconfiguration with Prognostics Consideration*

At the middle and low levels, receding horizon control (RHC) has shown promise for control reconfiguration due to its ability to handle constraints and changing model dynamics systematically [5]. Let us, without loss of generality, consider actuator faults as an illustrative example. Control actuator failures can be handled naturally in a RHC framework via changes in the input constraints and internal model.

Prognostic information can be integrated into the RHC problem as long as fault propagation is properly modeled and reflected in the state space model used for controller synthesis. For example, typically the receding horizon optimal control is derived by solving the following optimization problem

$$\min J=\int_{t_0}^{t_f}\frac{1}{2}[x^TQ_xx+u^TQ_uu]dt \tag{4}$$

subject to system dynamics constraints, actuation saturation, etc. In (4), u is the control to be solved, x is an augmented state vector, Q_x and Q_u are symmetric positive semi-definite matrices that can be designed to reflect the tradeoff between controller performance and control usage. Using a linear system state space model as example (the methodology can be extended to non-linear system though), control actuator prognosis (to stay with the illustrative example) can be introduced as a time varying *control effectiveness coefficient* matrix Λ,

$$\dot{x}=Ax+\Lambda Bu \tag{5}$$

where $\Lambda=diag(\lambda_1,...,\lambda_m)$, and λ_i is the control effectiveness coefficient of the i^{th} actuator with 1 being healthy and 0 being stuck. The prognostic modules estimate the evolution of λ_i over time, i.e. $\lambda_i(t)$.

Moreover, if the prognosis of the i^{th} actuator indicates that the actuator may get stuck in the near future, by constraining the movement of the actuator, the RHC controller can make sure that the actuator fails at (or close to) a preferable position so that its impact to subsequent control reconfigurations is minimal. The constraints can be formulated as,

$$\begin{aligned} &u_0-du_0\le u_i(t)\le u_0+du_0 \\ &\Delta u_0^l\le \dot{u}_i(t)\le \Delta u_0^h \end{aligned} \tag{6}$$

where du_0 is the reduced range for the i^{th} actuator around the favorable position (typically the neutral position) u_0, and is a function of the remaining useful life estimate. Placing constraints on $\dot{u}_i$ can also help to mitigate or defer actuator failure by avoiding aggressive control signals. False alarm is mitigated by relaxing the constraints upon receiving subsequent prognostics update. This failure prevention strategy was implemented in the UAV case study that follows.

III. ACM+P System Case Study: An UAV Example

The generic ACM+P concept presented in previous section was applied to an unmanned rotorcraft, the GTMax, using a software-in-the-loop simulation environment [12]. Due to physical limitations and capabilities available on the platform, only a subset of the general methodologies was applied for proof of concept.

The GTMax (Fig. 4) is an automated version of Yamaha's RMax, a remotely controlled helicopter weighing 128 pounds empty with a 10-foot rotor diameter. The aircraft weighs approximately 160 pounds in its test configuration. The GTMax was a primary research platform for the Defense Advanced Research Projects Agency (DARPA) Software Enabled Control (SEC) and Heterogeneous Urban Reconnaissance, Surveillance and Target Acquisition Team (HURT) projects [14].

Fig. 4. The GTMax Autonomous Rotorcraft

The baseline flight control system employs an adaptive neural network and Pseudo-Control Hedging (PCH) in a feedback linearization scheme to provide precise reference model tracking [13]. Extensive flight testing has proven the competence of the nominal controller. Furthermore, due to its neural network based adaptation mechanism, the nominal controller is capable to stabilize the vehicle in the presence of certain classes of fault conditions. However, it is not capable of stabilizing the vehicle when one of the flight control actuators has been completely immobilized or significantly degraded.

A. Swashplate Collective Actuator Failure and Active RPM Controller

The control vector for a typical single main rotor helicopter includes four inputs: collective, δ_{coll}; lateral cyclic, δ_{lat}; longitudinal cyclic, δ_{lon}; and tail rotor pitch, δ_{tr}. Helicopter vertical thrust is a strong function of both δ_{coll} and Ω, the angular rate of the main rotor. In the nominal state, vertical thrust is controlled by δ_{coll} with Ω held constant. A throttle control loop manipulates the throttle, δ_t, to maintain the speed of the main rotor, Ω, at a constant value, Ω_{com}.

The fault mode simulated for this study is a stuck collective actuator (δ_{coll}) failure. An active RPM controller that can accommodate a stuck collective actuator failure has been developed and successfully flight-tested on GTMax [11]. Under reconfigurable flight control, the RPM controller serves as an outer loop for the baseline throttle loop. It generates Ω_{com}, and the baseline throttle loop generates δ_t. Frequency separation allows the RPM controller and the baseline outer loop controller to operate simultaneously with δ_{coll} operating at a higher frequency than Ω.

The work presented in this paper is built upon the aforementioned baseline controller and the active RPM controller. The contribution of our work is the use of prognosis in control reconfiguration that involves reconfiguration of the baseline controller. Control restructuring in the event of collective actuator malfunctions does not include any reconfiguration of the baseline controller in [11], and with only the baseline and active RPM controllers, the system may still fail if the collective actuator got stuck at a very low power setting. With the ACM+P system, such catastrophe can be avoided.

Other possible failure modes that can be simulated on the GTMax software and hardware platform include swashplate actuators failures, tail rotor actuator failure, and various sensor faults. Since this research is focused on ACM instead of PHM, the diagnosis and prognosis for the actuator are simulated. Over the past few years, there have been substantial interest and investment in flight actuator diagnosis and prognosis, especially for Electro-Mechanical Actuators (EMAs). Interested reader can refer to literature [15], [16] and [17] for more information.

B. ACM+P System for GTMax

Fig. 5 shows an ACM+P implementation for GTMax UAV. The left part of the hierarchy is the set of baseline systems that are already deployed on the UAV, and the right part represents the ACM+P system. In nominal operation, the mission planner assigns waypoints and associated parameters (such as maximal velocity, acceleration, and jerk, etc.) to the trajectory generator, which generates a vehicle flight path for the baseline flight controller. The baseline flight controller calculates the appropriate control commands and sends them to flight actuators and engine at the lowest level. Diagnostics and prognostics from onboard PHM modules triggers ACM+P system, and re-planning and reconfiguration can take place at all four levels as shown in Fig. 5.

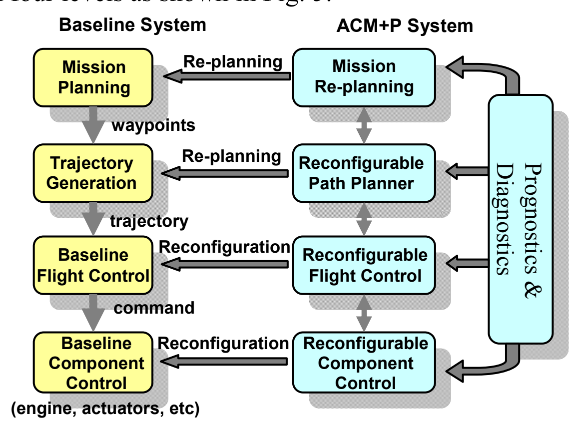

Fig. 5. ACM+P System for GTMax

Compared to a reactive FTC or ACM system, the ACM+P system introduces a new operational mode to handle prognostics. A simplified state flow model that can be applied to all four levels of reconfiguration/re-planning is shown in Fig. 6. When prognostic information is received, the ACM+P system switches to a *Prognostics-enhanced Failure Prevention Mode*, in which the fault situation is evaluated with consideration of performance requirement. Optimal fault mitigation control actions are applied to mitigate future fault. The optimization and decision making in ACM+P system must be capable of handling the uncertainties associated with prognosis and the system should return to Nominal Mode once a false alarm is cleared.

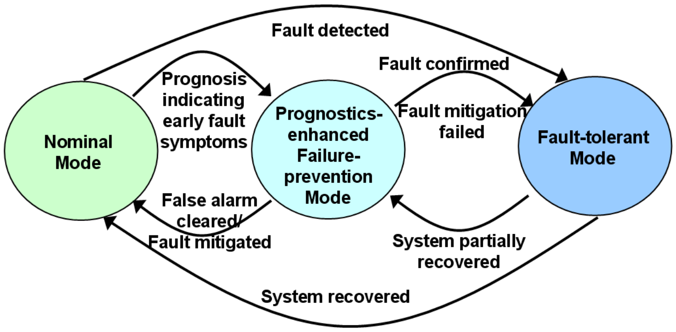

Fig. 6. Prognostics-enhanced Control Reconfiguration State Flow Model

C. *Prognostics-enhanced Failure Prevention Flight Controller*

For the stuck collective actuator failure mode, the actuator prognosis gives an estimate of when it is going to fail (with confidence bounds for that estimation). The information is utilized to prevent the collective actuator from getting stuck in a low power setting by restricting the range of its movement about a trim (hover) condition, thus a catastrophic failure can be avoided. The Failure Prevention Controller adjusts collective actuator magnitude limits based on the prognostic information and activates a secondary main rotor RPM controller. The secondary RPM controller employs a simple PID mechanism and provides two benefits: 1) it partly compensates for the performance degradation caused by collective actuator movement constraints; 2) it helps to stabilize the vehicle when the collective actuator fails unexpectedly especially during the unsafe time period between the occurrence of the failure and the time it is detected by the diagnostic module. Fig. 7 shows the architecture of the *Prognostics-enhanced Failure Prevention Controller* implemented for the simulated collective actuator failure.

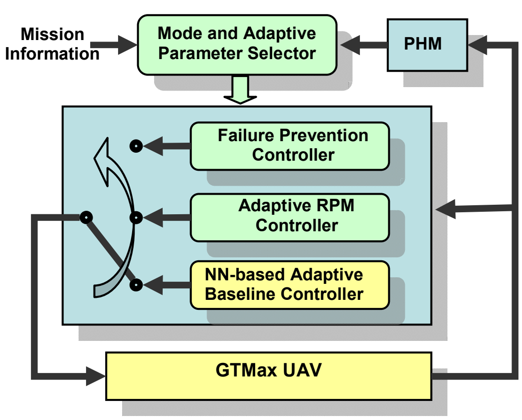

Fig 7. Prognostics-enhanced Failure Prevention Flight Controller (NN stands for Neural Networks)

The actuator movement limits are a function of remaining useful time (RUL), RUL confidence bounds and performance requirement. At the current stage, a fuzzy logic based system is utilized to determine the limits. Prognostic uncertainty is represented by a cumulative distribution function (CDF) which describes the probability of failure (PoF) as a function of time. The architecture of the fuzzy logic actuator constraint controller is shown in Fig. 8. The controller has two inputs, namely the risk level and the performance requirement. The risk level is evaluated by the prognostic module which provides prognostics at a possibly uneven interval (depending on the implementation of the prognostic routine and available onboard computational resources). The performance requirement is determined based on the type of maneuvers, velocity and acceleration requirement along future trajectories. It may also take into account of environmental factors, such as weather condition (e.g. wind/gust) and the possibility of encountering moving obstacles and avoiding hostile attacks.

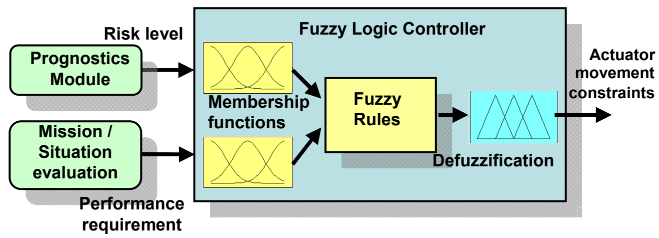

Fig 8. Fuzzy Logic Controller for Actuator Movement Constraints

One practical way to represent the risk level is to use the PoF value at the evaluation time (the point of time in the future when failure risk is evaluated). Proper choice of evaluation time allows the reconfigurable control strategies to be executed prior to the occurrence of the failure. The PoF value can be obtained from RUL CDF as shown in Fig. 9. A typical fuzzy rule takes the following form, "if risk level is HIGH and performance requirement is MID, then set actuator movement constraint to TIGHT".

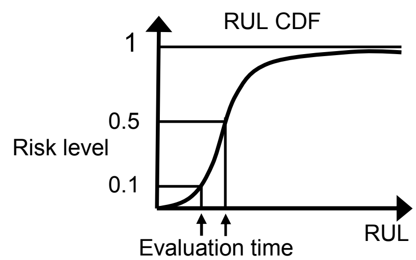

Fig 9. Determining Risk Level Using the RUL CDF

Fig. 10 shows the simulation results of the Fuzzy logic controller using simulated prognostics and a typical UAV mission (90 hours duration). Random noise was added to the prognosis to simulate prognostic uncertainty. The controller produced an output every 6 minutes. It can be observed that the actuator movement was confined to about 15% of original full range at the end of the simulation when the prognostics indicated a PoF of about 0.6. The spikes in the controller output were caused by the prognostic uncertainty, which can be considered false alarms in the system.

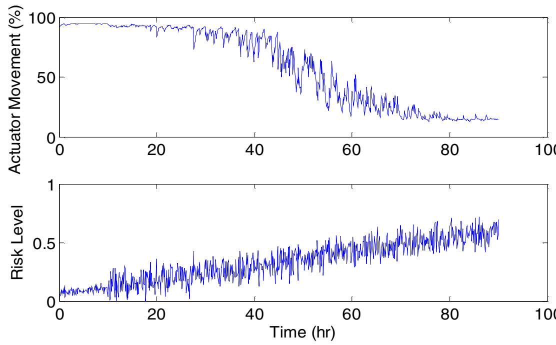

Fig 10. Fuzzy Logic Controller Output with Simulated Prognostics

The prognostics-enhanced failure prevention control reconfigures the baseline flight controller by adjusting the actuator magnitude limits. The actuator model used for Pseudo-Control Hedging purposes in the baseline controller has form [13],

$$\dot{\hat{\delta}} = \lim_{\lambda \to +\infty} sat(\lambda(sat(\delta_{des}, \delta_{\min}, \delta_{\max}) - \hat{\delta}), \dot{\delta}_{\min}, \dot{\delta}_{\max}) \qquad (3)$$

where $\delta = [\delta_{coll}, \delta_{lat}, \delta_{lon}, \delta_{tr}]$, $\hat{\delta}$ is the actuator position estimates, δ_{des} is the desired actuator positions, $\delta_{\min}, \delta_{\max}$ are the actuator magnitude limits and were set to the following units in the baseline controller.

$$\begin{aligned} \delta_{\min} &= [-2.5, \quad -1, \quad -1, \quad -1] \\ \delta_{\max} &= [1, \quad 1, \quad 1, \quad 1] \end{aligned} \qquad (4)$$

These units map to the full range of main rotor blade pitch, lateral tilt and longitudinal tilt of the swash plate, and tail rotor blade pitch available to the human safety pilot.

The simulation presented in this paper adjusts the limits of collective actuator only. The other limits can be reconfigured to accommodate swashplate actuator failure and tail rotor actuator failure in a similar way.

For comparison purpose, Fig. 11 shows the simulation result of a reactive ACM system with only the baseline controller and the RPM controller. The GTMax was executing an aggressive e-turn maneuver during which the collective actuator was stuck at a very low power setting (-1.1) at time 5680. The RPM controller was activated but still failed to maintain altitude (subplot 1) even running at a saturated high value (950 RPM in subplot 2).

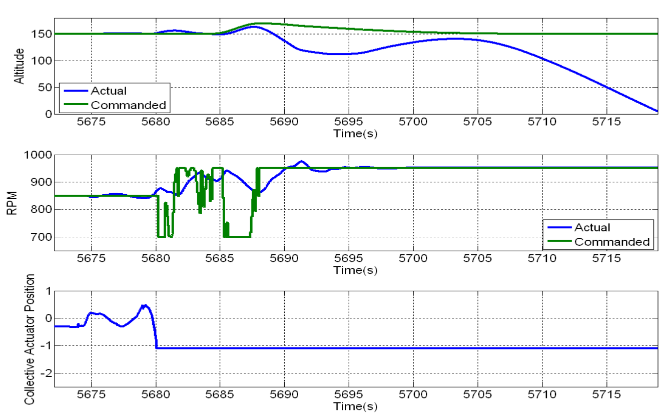

Fig. 11. Simulation results with reactive ACM. Subplot 1 (from top): altitude in feet. Subplot 2: main rotor RPM, and subplot 3: collective actuator position.

The same maneuver and failure mode were simulated with ACM+P system and the results are shown in Fig. 12. In this case, the Failure Prevention Control has been activated before the occurrence of the failure and the actuator magnitude limits were set to [-0.8, 0.1] as shown in subplot 5. The actuator got stuck at position -0.6 at time 580. The ACM+P system switched control to RPM control at time 582. With the ACM+P system, the UAV managed to complete the mission with acceptable position tracking error in east and north direction (subplot 2 and 3), but a considerable error in altitude tracking (subplot 1).

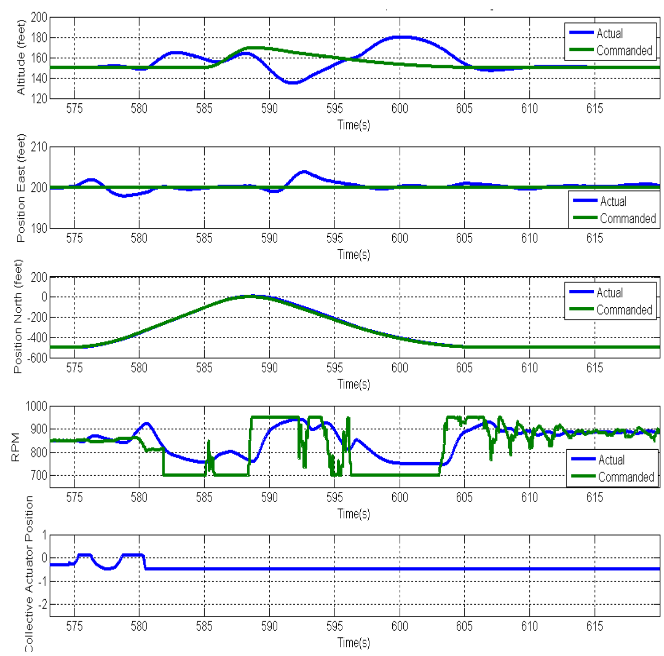

Fig. 12. Simulation results with ACM+P System; subplot 1: altitude in feet; subplot 2: position in feet (east); subplot 3: position in feet (north); subplot 4: main rotor RPM; subplot 5: collective actuator position.

D. *Mission Re-planning*

Typically, UAV missions are represented as a set of waypoints and performance parameters (e.g. maximal velocity, acceleration, and jerk) specified for the maneuvers between the points. Adjustment of mission parameters will affect the flight path generated for the flight controller and eventually affect the execution of the mission both in terms of time and fuel consumption. Based on diagnostic and prognostic information, ACM+P mission re-planning module solves the optimal mission parameters that fulfills (possibly relaxed) mission requirement subject to both current and future performance constraints.

Fig. 13 shows the simulation results with mission adaptation for the same failure scenario. In addition to the *Prognostics-enhanced Failure Prevention Controller* discussed above, a mission adaptation module adjusted the e-turn maneuver parameters (velocity, acceleration and jerk) to reduce the aggressiveness of the maneuver after the occurrence of the fault. As a result, it takes longer time to finish the mission (45 seconds compared to 40 seconds in Fig. 13, and 20 seconds in Fig. 12). However, the altitude tracking error is noticeably better with a maximal altitude tracking error of 8 feet compared to about 25 feet in Fig. 13.

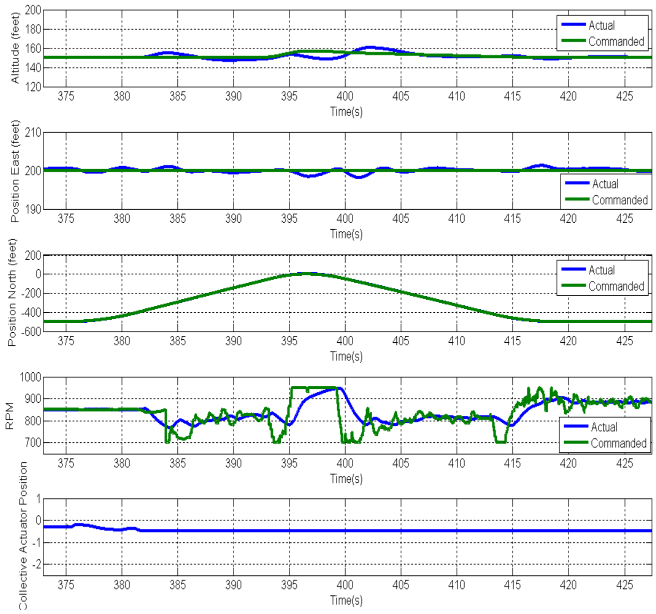

Fig. 13. Simulation results with mission adaptation; subplot 1: altitude in feet; subplot 2: position in feet (east); subplot 3: position in feet (north); subplot 4: main rotor RPM; subplot 5: collective actuator position.

IV. Potential Applications of ACM+P System

The case study presented in this paper is based on an autonomous rotorcraft platform. However, the generic ACM+P architecture and methodologies are also applicable to fixed wing aircraft. For example, similar to using RPM controller to accommodate collective actuator failure on a rotorcraft, the engine thrust control can be utilized to accommodate elevator actuator failure on a fixed-wing aircraft. Most commercial and military aircraft are developed with fly-by-wire flight control systems and are implemented with various degrees of redundancy in terms of controls, sensors and computing. The middle and low levels of the ACM+P system can be utilized to mitigate potential catastrophic failures in these systems.

The high level ACM+P modules (mission planning, resource/load distribution, etc.) using long term prognostic information is more important for an autonomous system designed to execute a long mission with little human intervention, such as a deep space probe. Automated fault accommodation and prevention with consideration of prognostic information are critical to future autonomous space systems.

V. Conclusions & Future Work

This paper presented a hierarchical proactive automated contingency management system that automatically accommodates and mitigates faults in an advanced autonomous system using diagnostic and prognostic information. The main contributions of this paper are:

1) a new category of FTC system, namely proactive FTC system, was introduced;

2) a hierarchical ACM+P architecture and generic methodologies were developed; the architecture allows faults to be accommodated at various levels using component level control reconfiguration, system level control reconfiguration and mission re-planning capabilities;

3) using prognostics in control reconfiguration and mission re-planning for automated fault accommodation and prevention was investigated and evaluated on an autonomous rotorcraft system, and simulation results were presented.

Several interesting research areas have been identified to further develop the ACM+P concept and eventually integrate it into real world autonomous systems.

1) specific algorithms for component level control reconfiguration were not addressed in this paper. Component level FTC using diagnostics has been studied over the past decade, but prognostics has to our knowledge previously not been taken into consideration in the control loop. The consideration of prognostics in component level control reconfiguration provides a unique opportunity to manage component life via control actions. The authors are currently working on this problem using an EMA flight actuator as an example.

2) in order to provide a truly comprehensive level of integrated fault isolation and accommodation for potentially hundreds of fault scenarios that could be encountered in flight requires a dynamic, model-based fault contingency management concept that can make decisions “on the fly”. System degradation and performance limit need to be identified online to accommodate these fault scenarios.

3) prognosis uncertainty management is an important issue that affects the performance of a Prognostics-enhanced ACM system significantly. In practice, accurate and precise prognostics has proven rather difficult to accomplish, thus uncertainty management and reduction techniques have to be implemented before the prognostics can be optimally utilized by the ACM+P system. Many accomplishments have been reported but major challenges still remain to be addressed [18]. Impact Technologies is actively involved in this research area and is developing a Prognosis Uncertainty Management and Analysis Toolbox to support both on-line and off-line prognosis uncertainty management.

Furthermore, in order for ACM+P systems to be used in safety-critical aerospace applications, they must be proven to be highly safe and reliable. Rigorous methods for ACM+P system verification and validation (V&V) must be developed to ensure that ACM+P system software failures will not occur, to ensure the system functions as required, to eliminate unintended functionality, and to demonstrate that certification requirements can be satisfied [20].

Acknowledgment

The GTMax simulation environment was developed by Dr. Eric Johnson and his colleagues at Georgia Institute of Technology. The active RPM controller and some generic UAV Fault Tolerant Control architecture and concept are an extension of the work of Dr. Graham Drozeski during his Ph.D. study at Georgia Institute of Technology. The authors also acknowledge the anonymous reviewers for their comments and suggestions.

REFERENCES

[1] Y. M. Zhang and J. Jiang, “Bibliographical Review on Reconfigurable Fault-Tolerant Control Systems,” in Proc. of the *5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Washington, D.C., USA, June 9-11, 2003, pp. 265-276.

[2] D. Shore and M. Bodson, “Flight Testing of a Reconfigurable Control System on an Unmanned Aircraft,” *AIAA Journal of Guidance, Control, and Dynamics*, vol. 28, no. 4, pp. 696-707, 2005.

[3] J. S. Litt, K. J. Parker, and S. Chatterjee, “Adaptive Gas Turbine Engine Control for Deterioration Compensation due to Aging,” Technical Report TM 2003-212607, NASA Glenn, Lewis, OH. 2003.

[4] L. Tang, M. J. Roemer, G. J. Kacprzynski, and J. Ge, “Real-Time Health Estimation And Automated Fault Accommodation For Propulsion Systems,” in *Proceedings of ASME Turbo Expo, Power for Land, Sea and Air*, May 14-17, Montreal, Canada. 2007.

[5] J. F. Monaco, D. G. Ward, and A. J. Bateman, “A Retrofit Architecture for Model-Based Adaptive Flight Control,” in Proc. *AIAA 1st Intelligent Systems Technical Conference*, Chicago, Illinois. September 20 - 22, 2004.

[6] L. Tang, G. J. Kacprzynski, M. J. Roemer, G. Vachtsevanos, and A. Patterson-Hine, “Automated Contingency Management Design for Advanced Propulsion Systems,” in *AIAA Infotech@Aerospace Conference and Exhibit*, Arlington, Virginia. September 26 - 29, 2005.

[7] M. J. Roemer, L. Tang, G. J. Kacprzynski, J. Ge, and G. Vachtsevanos, “Simulation-Based Health and Contingency Management,” in Proceedings of *IEEE Aerospace Conference*, Big Sky, Montana. Mar. 4-11. 2006.

[8] L. Tang, G. J. Kacprzynski, K. Goebel, J. Reimann, M. E. Orchard, A. Saxena, and B. Saha, “Prognostics in the Control Loop,” in Proceedings of *the 2007 AAAI Fall Symposium on Artificial Intelligence for Prognostics*, November 9-11, 2007, Arlington, VA.

[9] J. R. Bock, T. Brotherton, P. Grabill, D. Gass, and J. A. Keller, “On False Alarm Mitigation,” in Proceedings of *the IEEE Aerospace Conference*, Big Sky, MT, March 4-11, 2006.

[10] N. S. Clements, Fault Tolerant Control of Complex Dynamical Systems. PhD thesis, Georgia Institute of Technology, School of Electrical and Computer Engineering, Atlanta, GA 30332, April 2003.

[11] G. R. Drozeski, A Fault-Tolerant Control Architecture for Unmanned Aerial Vehicles, PhD thesis, Georgia Institute of Technology, School of Electrical and Computer Engineering, Atlanta, GA 30332, December 2005.

[12] E. N. Johnson, A. A. Proctor, J. Ha, and A. R. Tannenbaum, “Development and Test of Highly Autonomous Unmanned Aerial Vehicles,” *AIAA Journal of Aerospace Computing, Information, and Communication*, Vol. 1, No. 12, pp. 485-501, December 2004.

[13] E. N. Johnson, and S. K. Kannan, “Adaptive Trajectory Control for Autonomous Helicopters,” *AIAA Journal of Guidance, Control, and Dynamics,* Vol. 28, No. 3, pp. 524-538, May/June 2005.

[14] E. N. Johnson, D. Schrage, and G. Vachtsevanos, “Rotary wing final experiments for the software enabled control program," in *Proceedings of the American Helicopter Society 61st Annual Forum,* Grapeview, TX, June 1-3, 2005.

[15] C. S. Byington, M. Watson, D. Edwards, and P. Stoelting, “A Model-Based Approach to Prognostics and Health Management for Flight Control Actuators,” in Proc. *IEEE Aerospace Conference*, Big Sky, MN. March, 2004.

[16] C. S. Byington, M. Watson, and P. Kalgren, “A Unifying Approach towards Mechanical and Electrical Prognostics on Actuation Systems,” in Proc. *Integrated Systems Health Management Conference*, Cincinnati, OH, August. 2007.

[17] C. S. Byington and M. Watson, “Techniques and Engineering Software for Prognostics and Health Management of Flight Control Actuators,” Presented at *Aerospace Controls and Guidance Systems Conference*, October 20, 2005.

[18] K. Goebel, and N. Eklund, “Prognostic Fusion for Uncertainty Reduction,” in Proc. *AIAA Infotech@Aerospace Conference and Exhibit*, Rohnert Park, CA. 7-10 May, 2007.

[19] A. Saxena, M. E. Orchard, B. Zhang, G. Vachtsevanos, L. Tang, Y. Lee, and Y. Wardi, “Automated Contingency Management for Propulsion Systems,” in Proc. *European Control Conference*. Kos, Greece 2-5 July, 2007.

[20] L. Tang, A. Saxena, M. E. Orchard, G. J. Kacprzynski, G. Vachtsevanos, and A. Patterson-Hine, “Simulation-based Design and Validation of Automated Contingency Management for Propulsion Systems,” in Proc. *IEEE Aerospace Conference*, Big Sky, Montana. Mar. 3-10, 2007.

[21] S. Chatterjee and J. S. Litt, “Online Model Parameter Estimation Of Jet Engine Degradation For Autonomous Propulsion Control”, in Proc. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Austin, Texas, 11-14 August 2003.