# Proposal of a model-based fault identification genetic technique for more-electric aircraft flight control EM actuators

Matteo D. L. Dalla Vedova[1], Alfio Germanà[2] and Paolo Maggiore[3]

[1,2,3]*Politecnico di Torino, Torino, 10129, Italy*

*matteo.dallavedova@polito.it*
*alfio.germana89@gmail.com*
*paolo.maggiore@polito.it*

## ABSTRACT

In the last years, Electro-Mechanical Actuators (EMAs) are gradually replacing the older type of actuators based on the hydraulic power. In order to detect incipient failures due to a progressive wear of a primary flight command EMA, prognostics could employ several approaches; the choice of the best ones is driven by the efficacy shown in failure detection, since not all the algorithms might be useful for the proposed purpose. In other words, some of them could be suitable only for certain applications while they could not give useful results for others. Developing a prognostic algorithm able to identify the precursors of the above mentioned EMAs faults and their degradation pattern is thus beneficial for anticipating the incoming failure and alerting the maintenance crew such to properly schedule the servomechanism replacement.

The goal of this paper is to propose an innovative model-based fault detection and identification (FDI) method, based on Genetic Algorithms (GA), able to identify symptoms alerting that an EMA component is degrading and will eventually exhibit an anomalous behavior; in particular four kinds of EMA progressive fault are considered: friction, backlash, coil short circuit and electronics fault of controller. To assess the effectiveness of the proposed technique, an appropriate simulation test environment was developed: in particular, two MATLAB Simulink models representing the real EMA and the corresponding monitor have been used to simulate failures and evaluate the accuracy of the FDI algorithm. The results showed an adequate robustness and confidence was gained in the ability to early identify an eventual EMA malfunctioning with low risk of false alarms or missed failures. This paper aims to be a starting point to future works based on this method for PHM applications.



## 1. INTRODUCTION

Actuators are devices conceived to convert power from various sources (mechanical, electrical, hydraulic, or pneumatic) into motion. Such conversion is commonly used to operate flight control surfaces and several utility systems. Some of these actuators are safety critical; redundancy is the main (and obliged) option to reduce risk; moreover, components are required to be highly reliable. Rigorous programs of scheduled maintenance should guarantee that the system operates always in safety conditions. By the way, extreme (and possibly not expected) operative scenarios may lead to damage and unscheduled maintenance, with increased risk and costs, and possibly impact on mission. Monitoring functional parameters from the system of interest, it is possible to determine if an anomalous behavior is starting to occur at an early stage. It is also possible to determine the source of such anomalous behavior. The prediction of this kind of failures should be guaranteed at a high level of reliability. The discipline aimed to do so is called Prognosis and Health Management (PHM) (as reported by Byington, Watson, Edwards, & Stoelting, 2004); the application of the PHM strategies typically requires monitoring a set of system parameters in the form of electric signals. As a consequence, the application of PHM is favored on electrical systems, where no additional sensor is required. Prognostics are typically related to mechatronic systems having a complex non-linear multidisciplinary behavior; therefore literature proposes a wide range of fault detection and identification (FDI) strategies. Among these, it is possible to mention model-based techniques based upon the direct comparison between real and monitoring system (as proposed by Borello, Dalla Vedova, Jacazio, & Sorli, 2009), on the spectral analysis of well-defined system behaviors performed by Fast Fourier Transform FFT (Dalla Vedova, Maggiore & Pace, 2014), on appropriate combinations of these methods (Maggiore, Dalla Vedova, Pace & Desando, 2015) or on Artificial Neural Networks (as shown by Battipede, Dalla Vedova, Maggiore & Romeo, 2015).

As regards the onboard flight command actuation systems, it must be noted that Electromechanical Actuators (EMAs) are gradually replacing the older type of actuators based on the hydraulic power; these actuators are essentially electric motors which transfer rotational or translational power to the control surfaces by means of gearings. As they are relatively recent for aeronautical applications, researchers are evaluating efficient and effective methods to detect incipient failures in order to improve their functionality and safety. However, with respect to the older hydromechanical or electrohydraulic actuators, EMAs are typically more favorable to be monitored with a PHM methodology, given that the same sensors used to the control scheme and system monitors can often be used also for PHM analysis (i.e. no additional sensors are required). For these reasons, the study presented in this paper considers electromechanical actuation systems, according to the “More Electric Aircraft” paradigm (Quigley, 1993) and the “All Electric Aircraft” paradigm (Howse, 2003). For completeness, it must be noted that the concepts and the results reported in this paper about the design of reliable and fast prognostic FDI routines belong to a wider research activity focused on the diagnosis model-based approach and, in particular, on the parametric estimation task.

## 2. AIMS OF WORK

In this paper a new FDI algorithm based on the Genetic Algorithms (GA) is proposed, optimized and then validated through the comparison between predicted values and the behavior of a numerical EMA virtual test-bench, conceived and modeled for the purpose. In order to evaluate the accuracy of the prediction at the different conditions and to assess the field of validity of the proposed method, different combinations of progressive faults have been considered. In particular, according to hypothesis shown by Maggiore et al. (2015), authors evaluate the following progressive faults: BLDC motor stator coil short circuit, backlash and dry friction acting on the mechanical transmission and drift of the gain of the closed-loop position control logic.

## 3. EMA REFERENCE MODEL

As shown in figure 1, a typical EMA is composed of:

1. an actuator control electronics (ACE) that closes the feedback loop, by comparing the commanded position (FBW) with the actual one, elaborates the corrective actions and generates the reference current Iref;
2. a Power Drive Electronics (PDE) that regulates the three-phase electrical power;
3. an electrical motor, often BLDC (BrushLess Direct Current) type;
4. a gear reducer having the function to decrease the motor angular speed (RPM) and increase its torque to desired values;
5. a system that transforms rotary motion into linear motion: ball or roller screws are usually preferred to acme screws because, having a higher efficiency, they can perform the conversion with lower friction;
6. a network of sensors used to close the feedback loops (current, angular speed and position) that control the whole actuation system.

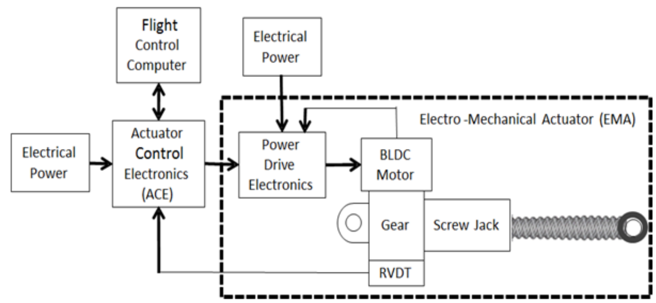



Figure 1. Electromechanical Actuator Scheme.

As previously stated, the primary goal of the research is the proposal of a technique able to identify symptoms alerting that an EMA is degrading: therefore, in order to assess the robustness of this technique, a suitable simulation test environment has been developed. The proposed numerical model, developed by Maggiore et al. (2015) and shown in Figure 2, is consistent with the EMA architecture reported in Figure 1 and it is implemented in MATLAB/Simulink® environment. Such a model is able to simulate the dynamic response of the real actuation system taking into account the effects due to aforesaid progressive faults, conversion from analogic to digital of the feedback signals (ADC), electrical noise acting on the signal lines and position transducers affected by an electrical offset.

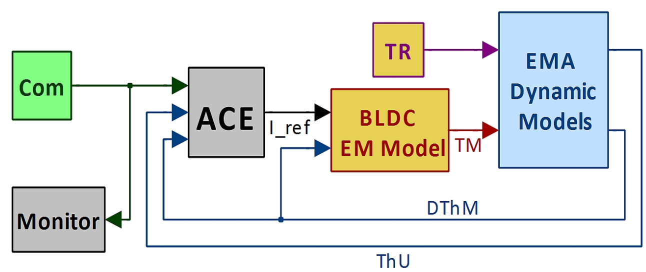



Figure 2. Proposed EMA block diagram.

It is composed by six different subsystems that will be briefly described in the followings.

1. an input block that generates the different position commands (Com);
2. a subsystem that, as shown by Todić, Miloš and Pavišić (2013), simulates the actuator control electronics, closes the feedback loops and generates as output the reference current $I_{ref}$ (ACE);

3. a subsystem simulating the power drive electronics and the trapezoidal BLDC electromagnetic model, that evaluates the torque developed by the electrical motor as a function of the voltages generated by the three-phase electrical power regulator (BLDC EM Model); this model, proposed by Maggiore et al. (2015), has been developed according to the mathematical models and the assumption reported by Çunkas and Aydoğdu (2010), Halvaei Niasar, Moghbelli and Vahedi (2009) and Hua and Zhiyong (2008);
4. a subsystem simulating the EMA mechanical behavior by means of a 2 degrees of freedom dynamical system (EMA Dynamic Models);
5. another input block simulating the aerodynamic torques acting on the moving surface controlled by the actuator (external forcing TR);
6. a block simulating the monitoring system (Monitor).

In order to validate the proposed numerical model, the dynamic response developed by the aforesaid system under certain operating conditions (control input, boundary conditions and entities of different faults) was compared with data obtained from literature. In particular, the back-EMF and phase currents waveforms, related to different values of the rotor angular velocity, and the dynamic responses of the BLDC motor, caused by various command inputs, have been compared with some experimental evidences and corresponding cases available in literature (Lee & Ehsani, 2003 and Kaliappan & Chellamuthu, 2012), highlighting a satisfactory compliance.

## 4. EMA FAILURES AND DEGRADATIONS

Since EMA have been only recently employed in aeronautics, their cumulated flight hours or on-board installations are not enough to allow a reliable statistics about the more recurring failures. Gökdere, Chiu, Keller & Vian (2005) show that it is possible to discern between four main categories of failures:

1. mechanical or structural failures;
2. BLDC motor failures;
3. electronics failures;
4. sensor failures.

The present work has been mainly focused on the combined effects of four different progressive faults: backlash ($BLK$) and dry friction ($F_{ST}$) acting on the mechanical transmission (as a consequence of wear phenomenon), turn-to-turn short circuits affecting the BLDC motor stator coils ($N_a$), and drift of the gain of the closed-loop position control logic ($G_{prop}$). These failure modes were selected, on the basis of information found in literature about the incidence and the criticality of different progressive faults (Kenjo & Nagamori, 2003, Chesley, 2011, and Weiss, 2014), in order to individuate a significant case of study.

As known, dry friction phenomena always occur when two surfaces are in relative motion: when friction coefficients increase due to wear, reaction torque becomes higher and the motor must provide higher torques to actuate the control surface. As shown by Borello and Dalla Vedova (2012), the increased dry friction, while still not causing the seizure of the entire system, reduces the servomechanism accuracy and, sometimes, influences the system dynamic response generating unexpected behavior (stick-slip or limit cycles). The mechanical wear also generates backlash in EMA moving parts such as gears, hinges, bearings and especially screw actuators. These backlashes, acting on the elements of the mechanical transmission, reduce the EMA accuracy and can lead to problems of stiffness and controllability of the whole actuator (Borello and Dalla Vedova, 2014). BLDC motor failures are mainly seen as progressive coil short-circuits, or bearing wear generating rotor static eccentricity. As shown by Shashidhara & Raju (2013), there is a consensus that 35-40 % of induction motor breakdowns could be attributed to the stator winding insulation. The short-circuit failure mode usually starts between a few coils belonging to the same phase (coil-coil failure). Since into short-circuited coils the voltage remains the same and the resistance decreases, a high circulating current arises, generating a localized heating in conductor: this heating favors the extension of the failure to the adjacent coils. If this kind of failure is not promptly detected it could propagate and generate phase-phase or phase-neutral damages. The progressive stator coil short-circuit effects have been modeled by means of a simplified numerical algorithm shown by Maggiore et al. (2015). Since both failures change the magnetic coupling between stator and rotor, the algorithm simulates these faultss modifying values and angular modulations of the back-EMF coefficients[1].

$$ke\_a = Ke_i \cdot Ce_i \cdot (1 + \zeta \cdot cos(\vartheta_r)) \qquad (1)$$

These constants ($ke\_a$, $ke\_b$, $ke\_c$) are used to calculate the corresponding back-EMF (*ea*, *eb*, *ec*) and mechanical couples (*Cea*, *Ceb*, *Cec*) generated by the motor phases. Sensor and electrical components are not less important than the other ones and their degradations are often characterized by rather fast temporal evolutions. However, considering suitable time scales, it is possible to evaluate precursors that can be used to take an action (Ginart, Brown, Kalgren & Roemer, 2007 and 2008). Wanting to test the effectiveness of the algorithm of FDI also proposed in the presence of this type of progressive failure (and, especially, in the case of interactions with other considered failures), the authors decided to examine a simple case of electronic fault caused by an unexpected drift of the gain of the closed-loop position control logic.

1 The proposed algorithm, implemented by means of the functions f(u) contained in the BLDC EM Model block diagram reported in figure 2, acts on the three back-EMF constants *Cei* (one for each branch) modulating their trapezoidal reference values $Ke_i$ as a function of coil short circuit percentage, static rotor eccentricity $\zeta$ and angular position $\vartheta_r$.

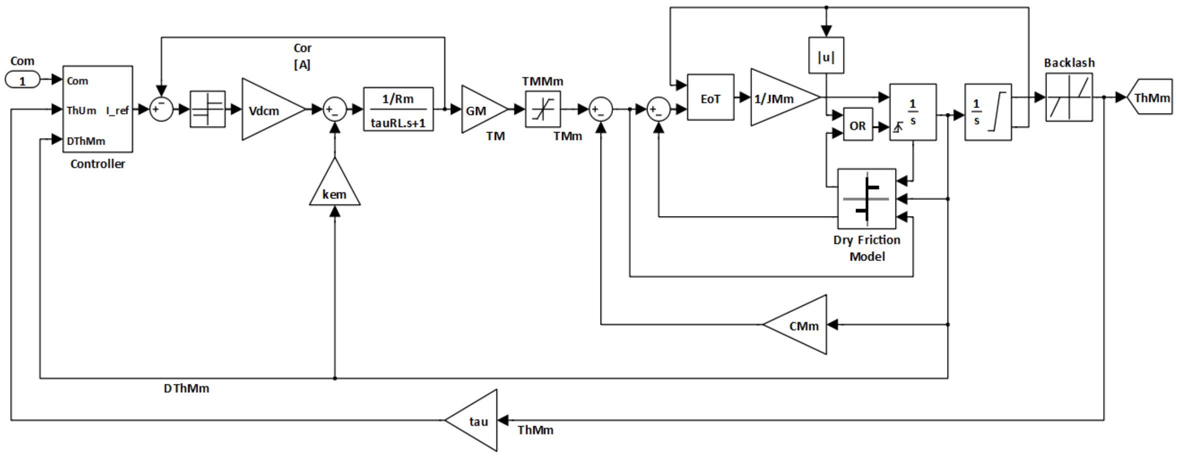



Figure 3. EMA Monitoring Model block diagram

## 5. EMA MONITOR MODEL

The above mentioned Simulink model, as explained in the previous section, is able to simulate the dynamic behavior of an actual electromechanical servomechanism taking into account the effects due to command inputs, environmental boundary conditions and several kinds of failure. So, even with proper limitations, this model allows simulating the dynamic response of the real system in order to evaluate the effects of different faults and designs, analyses and tests different diagnostic and prognostic monitoring strategies. In order to conceive a smart system able to identify and evaluate the progressive failures by means of a GA-based parameter optimization process, it is necessary to compare its dynamic behaviors with those provided by a simpler EMA model. To this purpose, authors developed a new Monitor Model (MM), representing a simplified version of the detailed EMA model, having the same logical and functional structure; such a model, with respect to the detailed one, is able to give similar performance, although less detailed, requiring less computational effort and reduced computational time. The MM has been run several times, during the GA-based optimization process, in order to calculate proper prognostic parameters used by the authors to perform the FDI of the detailed EMA model (i.e. the real system). It must be noted that the input of this model is the same as that used for the reference model, in order to make a comparison between these two models when studying the dynamic response. Obviously, it is important that the dynamic response of the monitor model must be as closer as possible to the reference model dynamic response.

## 6. FAULT DETECTION/IDENTIFICATION ALGORITHMS

Several optimization techniques are commonly used also for model parameter estimation tasks, which can be classified into two main categories: deterministic (direct or indirect) and probabilistic (stochastic, as Monte Carlo method, simulated annealing and genetic algorithms).

As reported by Dalla Vedova et al. (2014), a large part of these methods are local minima search algorithms and often do not find the global solution (i.e. they are highly dependent on a good initial setting). Local-minima approaches would not be robust and may provide a false indication of parameter changes in an on-line system (i.e. a wrong selection of starting settings could determinate problems of convergence or global minima). Otherwise, global search methods, such as genetic algorithms and simulated annealing, provide more promising options for on-line model identification (Raie & Rashtchi 2002, Alamyal, Gadoue & Zahawi 2013).

Genetic Algorithms (GAs) have been used in science and engineering as adaptive algorithms for solving practical problems and as computational models of natural evolutionary systems (Mitchell, 1996). About that, it must also be noted that, especially in order to implement a model-based FDI algorithm able to perform the health diagnosis of a real EMA evaluating several variables (typically five or more), the method based upon GAs are usually more effective and reliable with respect to other approaches (e.g. deterministic methods). In recent years the applications of genetic algorithms in the development of diagnostic systems based on numerical models have found wide interest in the scientific world and have led to several technical applications. In particular, in the field of mechatronics and electromechanical systems (with particular emphasis on electric machines), have been published many researches about new diagnostic and prognostic algorithms which integrate GAs optimization and model-based approach (Alamyal, Gadoue & Zahawi, 2013). For example, Raie & Rashtchi (2002) developed a parameter identification method, based upon genetic algorithm, for the detection and magnitude determination of stator turn-to-turn coil faults, based on a parameter identification of a model in which the turn fault is considered.

Starting from these considerations, in this work the authors have developed a new model-based FDI technique to identify fault levels of an EMA analyzing its dynamic response and comparing it, through a process of optimization (GAs), with the response generated from a numerical model. Then, the proposed method to detect these faults is based on the comparison of two signals coming from a reference system, and a monitor model. The former can be, for example, the angular speed or a current circuiting in a phase of a Brushless DC (BLDC) motor of a real EMA for which the fault detection is needed, or, as in the case of this work, of a modelled EMA able to simulate the examined faults. The latter is a simplified model with the requirement to be simple and fast in terms of implementation and computational time, since proposed method needs several iterations, making the heavily detailed model inappropriate. However, once determined the output to study, either of a real or a simulated EMA, and once modelled the monitor model, the comparison is performed by an optimization algorithm. This algorithm minimizes a quadratic error function by changing iteratively one or more parameters (defined as representative of the examined faults) of the monitor model until the output signal best overlaps with the reference system response. If the parameters calculated by the optimization algorithm match with the real ones, the method has worked properly; if the monitor model is accurate enough, the optimization algorithm gives a good detection of the system health.

As previously mentioned, the goal of this paper is to study, through Genetic Algorithms (GA), a detection method for some of the most representative faults EMA related. Two MATLAB Simulink models representing the real EMA and the monitor model were performed. To this purpose it must be noted that GAs are a class of evolutionary algorithms that take inspiration to the natural selection process. Optimization starts with a population of points (called chromosomes) which together represent the human genome. Each chromosome is a potential solution of the problem, the so called fitness function (the abovementioned error function), calculated for each of them. According to the obtained value, a rank is assigned to them: since it is a minimization, chromosomes who give lower fitness values have a better rank and are selected to be the parents of a new population of points (the following generation) created by means of different operators called crossover (a combination of parents), migration and mutation. This process is repeated iteratively until the last child of the last generation fulfill a stopping criterion, that can be a tolerance on the fitness function, a limit on the stall generations, a maximum number of generations, etc.

By tuning these settings, the method can be more or less fast or may or not converge to a final solution. It is important to consider that there is a strong dependence on the particular problem taken into account.

## 7. GENETIC ALGORITHM TUNING

Matlab gives the opportunity to set different GA options in order to improve the convergence and the speed of the algorithm. GAs convergence is characterized by a strong dependence from the treated problem so that there is not an universal configuration that works in every case, hence a setting used for a problem such as a single fault optimization is very different from a setting for a multiple fault optimization. First of all it is important to specify the type of function (fitness) that has to be set in order to make the optimization. As already seen, the typical function used for this kind of problems is the error function. Error function is made up with the square of the difference between the measured output of the EMA and the modelled one. When this error is minimized by the optimization algorithm, the two curves match at their best. As the monitor model is built to find a fault of the real EMA, it takes into account of different parameters that represent several typologies and levels of damage, one if a single fault simulation is performed, otherwise four in combined fault. The algorithm changes iteratively these parameters inside the Simulink monitor model until the error is minimized, giving as results parameter values that should approximate the real faults. In order to have a good optimization by GA, different parameters must be tuned.

### 7.1. BC definition

It is important to define the research domain by the boundary conditions. For every parameter it is known the operating range that includes the starting nominal condition and a hypothetical full damage limit (e.g. $N_a$ cannot be higher than 1 or less than 0.8, $F_{ST}$ cannot be less than 0.1689). This is important since the algorithm will search inside a limited space with a better probability to find the right solution with the same population size. In Table 1 are reported the boundary conditions for the examined faults.

Table 1. Progressive Faults Boundary Conditions.

| | $N_a$ [#] | $F_{ST}$ [$N$] | $BLK$ [$rad$] | $G_{prop}$ [$s^{-1}$] |
|---|---|---|---|---|
| *Lower Bound* | 0.8 | 0.1689 | 0 | 5e4 |
| *Upper Bound* | 1 | 0.8445 | 0.04 | 15e4 |

### 7.2. Population size

The population size represents the number of individuals handled in each iteration. A small population size could not give the diversity necessary to search in all the available space, while a too large population makes the computational time too high. This is due to the fact that when an individual is tested by the algorithm, its fitness is calculated by the Simulink monitor model.

By default, Matlab uses 20 individuals. These are enough for single fault parameter estimation where the function has a global minimum well defined. In contrast, for multiple faults the fitness function becomes more complex since the electric fault and the friction hide each other's. For this reason, in this last case, a bigger population is considered.

### 7.3. Initial population

By means of the initial population it is possible to define a group of possible solutions of the problem. It is impossible to define an initial population with all the possible combinations of failure but it is possible to define a vector with the solution for the nominal condition. This helps saving time when EMA is in a healthy condition; in fact the algorithm finds the global minimum at its first iteration without the inconvenient of a long computation.

### 7.4. Initial range

In order to better search into the available space it is important to give an initial range. The initial range is the dimension of the space in which the algorithm spreads its individuals. If the problem is constrained, it is a good choice to put as initial range the boundary values. The importance of the initial range resides into the diversity: if the initial range is too small, the search of the global minimum would be restricted to small parts of the domain. This makes the mean distance between individuals too small to make a good search with the consequence that the algorithm might get into a local minimum. The proper initial range value, instead, gives the right diversity, i.e. individuals are spread into the whole domain giving a better chance to get into the global minimum.

### 7.5. Fitness function scaling

The fitness function scaling is the way through the GA gives a value to the fitness function calculated for each individual in order to have a meter for choosing the parents. The fitness scaling adopted in this work is "rank", i.e. the selection is not based on individuals fitness value but by a rank expressed as a natural number dependent on the value of the fitness. To better understand this method, imagine having two individuals, the former with a fitness of 100 and the latter with a fitness of 1. Obviously, the second individual is much better since it is two orders smaller than the first one. If the choice of the parents was dependent on the fitness value, the probability to select the individual with fitness 100 would be very low with respect to the individual with fitness 1. Instead, with fitness scaling, the two individuals are labeled in a way so that the individual with a fitness of 100 has the rank "2" and the individual with fitness 1 has the rank "1". With this method the algorithm does not privilege those individuals that have a fitness value much more smaller than others, preserving a better diversity and making harder to get into a local minimum.

### 7.6. Selection function

The GA default selection function is stochastic uniform; according to this method, a line is lied out, with each parent corresponding to a section of length proportional to its scaled value (calculated by the fitness rank scaling). The algorithm moves along the line in steps of equal size. At each step, the algorithm chooses the parent from the section it lands on. The first step is a uniform random number that must be minor than the step size. This gives weaker members of the population (according to their fitness) a chance to be chosen and thus reduces the unfair nature of fitness-proportional selection methods.

### 7.7. Crossover fraction

The crossover fraction is a number between 0 and 1 that represents the percentage of children that will be produced by parents crossover. When crossover fraction is set to 1, all children are produced by crossover. This can make the algorithm faster in terms of convergence but the probability to get into a local minimum is higher. Instead, a crossover fraction equal to 0 means that all children are created by mutation, that makes the algorithm totally stochastic with the impossibility to converge when the minimum is caught.

### 7.8. Crossover function

It is known that the crossover can be performed in different ways: as reported by Germanà (2015), the best results for the considered application are achieved by the crossover heuristic. Once selected the two parents for reproduction, the algorithm draw a line between them. The children are created along this line in the opposite direction to the worst parent with a radius that can be specified by the user. This method makes the algorithm to converge faster than other crossover functions, at least for the considered problem. Mutation is performed by the adaptive feasible function that is ideal for constrained fitness function applications.

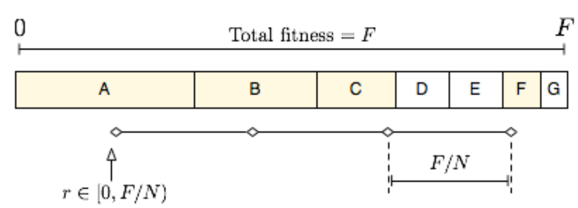



Figure 4. Selection stochastic uniform scheme

## 8. ANALYSIS OF THE PROPOSED FDI ALGORITHM

It is important to decide which faults and how many of them at same time one wants to simulate: it is clear that it is different if only a single fault occurs instead of two or many other faults. In fact in a real case, the user may not know which kind of damage is present in the considered EMA; hence the method must recognize the exact fault among a series of different parameters with the best accuracy.

On the other hand – if it is known which component gives problems – by a systematic study of the inputs and outputs of the system it is possible to find the hypothetical damage without testing a large amount of parameters. As reported by Germanà (2015), GAs are good for the search of a single fault, but in a real case may be necessary to find damage by searching among different parameters as there could be more than one fault. Differently from the single fault case, the multi-fault algorithm must be properly tuned since there are faults such as $N_a$ and $F_{ST}$ that hide each others (i.e. different damage combinations can lead to similar problem convergences). In fact, despite faults like $BLK$ and $G_{prop}$ have widely different effects on the dynamic response with an error function that has a global minimum univocally determined by any couple of these two parameters, for faults such as $N_a$ and $F_{ST}$the error function may bring to solutions far from the real state. Precisely, both affect the motor speed – and the user position – since a decrease of the former leads to an increasing regime speed while an increase of the latter causes a decrease of the same. This means that different couples of these parameters may give a best solution.

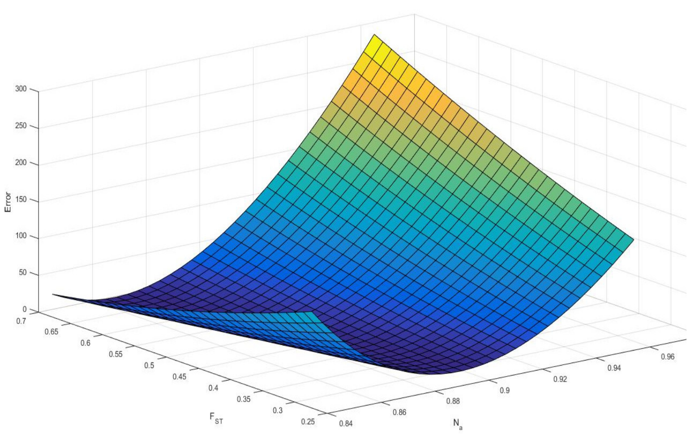



Figure 5. 3D representation of the fitness function (error) calculated for 1600 combinations of $N_a$ and $F_{ST}$.

This is better shown in figure 5 where the error function is given by a combination of 1600 couples of $N_a$ and $F_{ST}$ (the real values of the corresponding actual faults are $N_a$= 0.9 and $F_{ST}$.= 0.3378 N). As it is clear to see, lot of solutions are good (the valley labeled in blue). Moreover, the problem is accentuated by the fact that the motor speed error is much bigger with respect to the other output errors so that its effect on the optimization is predominant. This problem has been overcome modifying the algorithm by giving to all errors the same weight. In the single fault estimation, only one output is used to find the searched value; whereas, in the proposed multiple fault detection, the fitness function is made up with the sum of all the outputs, each one with its own scale. A variation of each parameter brings to the variation of all outputs in different ways but in certain cases these variations are several orders lower than others.

A normalization of all errors with their maximum value makes all functions included between 0 and 1 so that variations due to a change of a parameter gives an amount of error comparable for all outputs. Hence, since $F_{ST}$ causes a small variation also in the sinusoidal response, it is univocally determined without the influence of the motor speed in the step response (necessary to find the $N_a$ fault).

### 8.1. Multiple fault results

As mentioned earlier, was then carried out an extensive campaign of tests to evaluate the performance of the proposed method in various operating conditions (various types of controls, various combinations of type and level of the faults considered, sensitivity to electrical noise and to the different environmental conditions). In the following tables are described the damage type, extent and accuracy of the estimation showing also the difference between outputs before and after the optimization.

***Nominal condition*****:** a good fault detector must recognize also a non-damaged situation. In nominal condition the optimization algorithm must return the nominal parameters as reported in the table below. This case is the strongest with respect to the others since the proposed optimization algorithm is set to take into account as starting point the nominal condition parameters. This is achieved by defining in the initial population a vector containing these values so that when the algorithm starts its optimization process, it finds immediately the global minimum.

***Small damage*****:** usually, every system has a progressive fault evolution so that it is important for the algorithm to find a small variation in the model parameters. In this simulation a small percentage of the coil is short circuited, wearing causes a small backlash and friction, and finally no electronic fault is taken into account.

***Full damage:*** First of all it is appropriate to state that, in a case of real operation, the probability of occurrence of such a condition (i.e. very high fault levels for all types of progressive failure considered) is quite remote. In fact, progressive faults as wearing increase gradually so that they are detected before reaching too high values. But, taking into account that a robust and effective algorithm must recognize every fault situation in order to be considered a valid method, also this condition has been investigated. Results shown in Table 4 confirm that the FDI algorithm can predict also strongly nonlinear conditions.

***Random damage:*** finally, wishing to investigate the sensitivity of the proposed algorithm in more depth to different failure conditions (e.g. combinations of very heterogeneous fault), it has been developed a calculation routines capable of generating various fault combinations (setting the corresponding bounds and the desired level of heterogeneity) As reported in Table 5, the results accuracy is high as in the previous fault combinations so that the method is definitely effective.

Table 2. Results for a nominal condition combination.

| Fault | Estimation | True value | Accuracy |
|---|---|---|---|
| $N_a$ | 1 | 1 | 100% |
| $F_{ST}$ | 0.1689 | 0. 1689 | 100% |
| $BLK$ | 0 | 0 | 100% |
| $G_{prop}$ | 100000 | 100000 | 100% |

Table 3. Results for a small damage combination.

| Fault | Estimation | True value | Accuracy |
|---|---|---|---|
| $N_a$ | 0.96105 | 0.95 | 98.83% |
| $F_{ST}$ | 0.32989 | 0.3378 | 97.65% |
| $BLK$ | 0.00986 | 0.01 | 98.60% |
| $G_{prop}$ | 97984 | 100000 | 97.98% |

Table 4. Results for full damage combination.

| Fault | Estimation | True value | Accuracy |
|---|---|---|---|
| $N_a$ | 0.8141 | 0.8 | 98.23% |
| $F_{ST}$ | 0.8274 | 0.8445 | 97.97% |
| $BLK$ | 0.0403 | 0.04 | 99.25% |
| $G_{prop}$ | 149815 | 150000 | 99.87% |

Table 5. Results for random damage combination.

| Fault | Estimation | True value | Accuracy |
|---|---|---|---|
| $N_a$ | 0.961 | 0.95 | 98.41% |
| $F_{ST}$ | 0.6729 | 0.6756 | 99.6% |
| $BLK$ | 0.0098 | 0.01 | 98% |
| $G_{prop}$ | 124140 | 125000 | 99.31% |

## 9. CONCLUSIONS

Within this paper, we have demonstrated that, if properly set and calibrated, the proposed GA algorithm, used for FDI method, is very reliable in the identification of the increase of EMAs failures' precursors; in particular, GAs are suitable for parameters estimation since both single and multiple faults give accurate results for different levels of damage. We have tested the proposed method in various failures conditions and it proved to be effective for an operational scenario with a suitable time of execution (some minutes)[2]. In this regard, before concluding, it is however appropriate to make some observations. The first consideration is about the convergence: the problem can be represented by 1 to 4 variables and a fitness function constituted by the sum of 8 different outputs which guarantees univocally a global minimum when the right fault combination is applied to the variables of the problem. Accuracies are all over 90%, so we can conclude that the method converges appropriately. Moreover, it must be said that, even if the method is probabilistic, every simulation converges at the same result almost in the 100% of attempts, making the method suitable in terms of repeatability. The second consideration is about the convergence speed: when single fault estimation is performed, convergence happens in less time rather than in the multiple faults case, exactly approximately in 25 iterations. This is due to the simplified nature of the problem. In fact, for single fault parameter estimation, the fitness function is constituted only by one output out of 8 available, chosen according to the sensibility of this last to a variation of the faulty parameter. This makes the problem simple to be solved since exists only one value of the fault considered that returns the global minimum. Moreover, performing GA with only one parameter means to handle a smaller population, with the consequent lower computational time required. Usually, the time taken to converge in this case is about some tens of seconds. Evidently, this is different when multiple faults are considered. We chose to simulate 4 faults at the same time because only one output of the reference model is not sufficient to find univocally the global minimum. There are faults that yield similar effects to the same output, making the optimization harder. As a consequence, more than one output is considered, exactly 8 different outputs both for sinusoidal and step response. In this case the problem is more complex and the minimum is harder to find, so that more iterations are required to obtain the convergence. For this second problem, around 100 iterations are concerned. This is not the only reason of the slower convergence speed: in fact, in this case the optimization algorithm handles 4 different variables, with a consequent bigger population. For each iteration, fitness evaluation, fitness scaling, selection, crossover, mutation and migration are performed for every individual of the population, requiring more computational resources.

2 It must be noted that, at the moment, the time of convergence of the entire FDI process varies, as a function of number of variables and boundary conditions, from a few minutes to beyond half-hour. A so high elaboration time depends not only from the calculation process itself, but also from the limited performances of the calculator (a laptop equipped with a i5 processor) and from complexity of the SW performing the FDI. Obviously, these elaboration times are not acceptable for real-time processes (hypothesis that, however, we feel to refuse a priori), but they are not satisfactory also in case of FDI processes scheduled during the usual aircraft maintenance operations.

In conclusion, it must be noted that, in the context of prognostic applications, this FDI method can be considered satisfyingly reliable (also in case of combined failures or noisy signals) and it is possible to assess its validity even on other possible different conditions (i.e. various combination of progressive faults and boundary conditions); however, in order to extend its capabilities, it is our intention investigate further these issues in order to:

1. extend the method to a much larger number of progressive failure and boundary conditions;
2. verify the robustness and the convergence of this method also under particularly unfavorable conditions;
3. appropriately simplify the monitoring model and the optimization procedure so as to limit the corresponding processing times (limiting it to a few minutes or less).

## REFERENCES


Alamyal, M., Gadoue, S. M., & Zahawi, B. (2013). Detection of induction machine winding faults using genetic algorithm. *Diagnostics for Electric Machines, Power Electronics and Drives (SDEMPED)*, 9th IEEE International Symposium on, Valencia, pp. 157-161.

Battipede, M., Dalla Vedova, M. D. L., Maggiore, P., & Romeo, S. (2015). Model based analysis of precursors of electromechanical servomechanisms failures using an artificial neural network, *Proc. of the AIAA SciTech Modeling and Simulation Technologies Conference*, January, 5-9, Kissimmee, FL.

Borello, L., Dalla Vedova, M. D. L., Jacazio, G., & Sorli, M. (2009). A Prognostic Model for Electrohydraulic Servovalves, *Proc. of the Annual Conference of the Prognostics and Health Management Society*, San Diego, USA.

Borello, L., & Dalla Vedova, M. D. L. (2012). A Dry Friction Model and Robust Computational Algorithm for Reversible or Irreversible Motion Transmission. *International Journal of Mechanics and Control (JoMaC)*, vol. 13, n. 02, pp. 37-48, ISSN: 1590-8844.

Borello, L., Villero, G., & Dalla Vedova, M. D. L. (2014). Flaps Failure and Aircraft Controllability: Developments in Asymmetry Monitoring Techniques. *Journal of Mechanical Science and Technology (JMST),* vol. 28, v. 11, pp. 4593-4603.

Byington, C. S., Watson, W., Edwards, D., & Stoelting, P. (2004). A Model-Based Approach to Prognostics and Health Management for Flight Control Actuators, *IEEE Aerospace Conference Proceedings*, USA.

Chesley, J. C. (2011). *Handbook of Reliability Prediction Procedures for Mechanical Equipment*. USA Navy, Naval Surface Warfare Center, NSWC Carderock Division, Bethesda, Maryland.

Çunkas, M., & Aydoğdu, O. (2010). Realization of Fuzzy Logic Controlled Brushless DC Motor Drives using Matlab/Simulink. *Mathematical and Computational Applications*, vol. 15, n. 02, pp. 218-229.

Dalla Vedova, M. D. L., Maggiore, P., & Pace, L. (2014) Proposal of Prognostic Parametric Method Applied to an Electrohydraulic Servomechanism Affected by Multiple Failures, *WSEAS Trans. on Environment and Development*, vol. 10, pp. 478-490, ISSN: 1790-5079.

Germanà, A. (2015). *Genetic Algorithms for the prognostic of electro-mechanic servomechanisms for aeronautical applications*, MSc Thesis, Politecnico di Torino, Italy.

Ginart, A., Brown, D., Kalgren, P., & Roemer, M. (2007). On-line Ringing Characterization as a PHM Technique for Power Drives and Electrical Machinery. *Autotestcon, 2007 IEEE*, September 17-20.

Ginart, A., Brown, D., Kalgren, P., & Roemer, M. (2008). Inverter Power Drive Transistor Diagnostic and Extended Operation under One-Transistor Trigger Suppression. *Applied Power Electronics Conference and Exposition, 2008. APEC 2008.* February 24-28.

Gökdere, L. U., Chiu, S. L., Keller, K. J., Vian, J. (2005). Lifetime control of electromechanical actuators. *IEEE Aerospace Conference Proceedings*. March 5-12, Big Sky, MT. doi: 10.1109/AERO.2005.1559655

Halvaei Niasar, A., Moghbelli, H., & Vahedi, A. (2009). Modelling, Simulation and Implementation of Four-Switch Brushless DC Motor Drive Based On Switching Functions. *IEEE EUROCON 2009*. May 18-23, St.-Petersburg, Russia.

Howse, M. (2003). All-electric aircraft, Power Engineer, vol. 17, n. 4, pp. 35-37.

Hua, J., & Zhiyong, H. (2008). Simulation of Sensorless Permanent Magnetic Brushless DC Motor Control System. *Proc. of the IEEE International Conference on Automation and Logistics*, September, Qingdao, China.

Kaliappan, E., & Chellamuthu, C. (2012). Simplified Modeling, Analysis and Simulation of Permanent Magnet Brushless Direct Current Motors for Sensorless Operation. *American Journal of Applied Sciences*, vol. 9, n. 7, pp. 1046-1054. ISSN 1546-9239

Kenjo, T,, & Nagamori, S. (2003). *Brushless Motors: Advanced Theory and Modern Applications*, Sogo Electronics Press, Tokyo, Japan.

Maggiore, P., Dalla Vedova, M. D. L., Pace, L., & Desando, A. (2015). Proposal of fault analysis parametric method applied to an electromechanical servomechanisms affected by failures. *Intern. Journal of Prognostics and Health Management*, vol. 6, n. 1. ISSN: 2153-2648.

Mitchell, M. (1996). *An introduction to genetic algorithms*. Cambridge, MIT press.

Quigley, R. E. J. (1993). More electric aircraft. *Proceedings of Eighth Annual IEEE Applied Power Electronics Conference - APEC '93*, March 7-11, San Diego, CA.

Raie, A., & Rashtchi, V. (2002). Using a genetic algorithm for detection and magnitude determination of turn faults in an induction motor. *Electrical Engineering*, 2002, vol. 84, n. 5, pp. 275-279

Shashidhara, S.M., & Raju, P.S. (2013). Stator Winding Fault Diagnosis of Three-Phase Induction Motor by Parks Vector Approach. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering (IJAREIEE)*, vol. 2, n. 7.

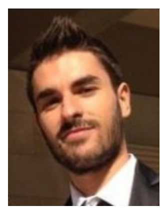
Todić, I., Miloš, M., & Pavišić, M. (2013). Position and speed control of electromechanical actuator for aerospace applications. *Tehnicki Vjesnik*, vol. 20, n. 5, pp. 853-860.

Weiss, J. (2014). Control Actuation Reliability and Redundancy for Long Duration, Underwater Vehicle Missions with High Value Payloads. *Proceedings of the 2014 Underwater Intervention Conference*.

## BIOGRAPHIES

**Paolo Maggiore** is a professor at the Mechanical and Aerospace Engineering Department of Politecnico di Torino, that joined in 1992, where he teaches aerospace general systems engineering. Currently his students are involved in projects ranging from hydrogen fuel cell powered airplanes and UAVs, and health monitoring of flight controls, to multi-disciplinary design optimization of aerospace systems design.

**Matteo D. L. Dalla Vedova** received the M.Sc. and the Ph.D. from the Politecnico di Torino in 2003 and 2007, respectively. He is currently assistant researcher at the Department of Mechanics and Aerospace Engineering. His research activity is mainly focused on the aeronautical systems engineering and, in particular, is dedicated to design, analysis and numerical simulation of on board systems and developing of prognostic algorithms for flight control systems and aerospace servomechanism.



**Alfio Germanà** received the M.Sc. in Aerospace Engineering from the Politecnico di Torino in 2015. With a great love for aircrafts, at the moment he is an engineer at HeadApp srl, a company which uses smartglass technologies for aeronautical and industrial applications. In particular, he is involved in the project "Eye4Fleet" for Aero Club fleet management.