

Advanced UAV Hybrid Simulator Model Based-on Dynamic Real Weather Data

Amr Abdelhamid, Peng Zong , and Bassant Abdelhamid

Abstract—In spite of all developed Unmanned Aerial Vehicle (UAV) simulators and all pilot training efforts, UAV crashes still exist. This is because of neglecting the effect of some environmental parameters such as wind turbulence during the UAV simulated training process. In this paper, a new software based simulator for advanced UAV flight training in dynamic real wind turbulence is introduced. The active hybrid architecture is utilized to exploit both VB.NET and MATLAB capabilities in the same simulation loop. A new implementation is introduced to exploit a free global weather forecast service for UAV simulated missions. The proposed simulator model can generate stochastic linear and angular wind velocities by integrating Dryden wind model with the UAV mathematical model based on real weather measurements. Encountering dynamic real weather parameters into the simulator model enhances both mission planning and simulated training procedures. The results show the ability of the proposed simulator to achieve innovative graphical capabilities, to satisfy the required system constraints, to provide realistic turbulence dynamics for the trainee, and to advance trainee experience through challenging real wind turbulence in the simulated training and fulfill the mission goals.

Index Terms—Software engineering, UAV systems and dynamics, active hybrid architecture, Dryden wind model.

I. INTRODUCTION

Over the last decades, aerial simulated training is approved as a replica for the real training, due to the high cost and complicated procedures of the latter [1], [2]. As a result, many simulators are developed using different commercial and open source packages with various fidelities considerations [3]. Such simulators targets different UAV aspects such as: mission planning [4], equipment simulation [5], Hardware-in-the Loop (HiL) simulation for different UAV algorithms [6]-[8], and operator training. For the latter, flight simulators plays a significant role in the Simulated Training Procedures (STPs). It is shown in [9], [10] statistics that UAV accidents occur in higher rate than commercial airlines due to fatal aircraft and UAV accidents . Moreover, the pilot errors contribute more than 50% from the total crash causes [11].

Owing the advent of high processing capabilities of recent multi-core processors, software based simulator gains much attention due its efficiency, reliability and its large Mean

Time Between Failure (MTBF). Therefore, ten workstations, network infrastructure, four Liquid Crystal Displays (LCDs), and a physical cabin are utilized to build a fixed based simulator. MATLAB and RT-LAB are used to develop flight simulation software and control loading model [12], [13]. In addition, at Sejong university, Sejong UAV FlightSim (SUF) is developed based on Commercial-Off-The-Shelf (COTS) computer hardware and software [14]. On the other hand, in [15], a low cost SCALAB virtual reality simulator is developed for Uh-1h helicopter. It employs three workstations and network infrastructure using FlightGear and costs 15000 USD. Finally, the authors in [1], [2] developed a standalone single workstation software simulator based on active hybrid architecture (VB.NET and MATLAB/Simulink) using the Component Object Model (COM) technology.

All aforementioned simulators in literature missed the effect of physical weather on the simulated remotely piloted missions. As a result, this paper is intended to account real weather conditions for UAV training simulators. Weather is considered one of the most stochastic phenomena that affects all aerial vehicles from the huge civilian aircrafts to UAVs. It could be deduced from the aircraft crashes information statistics in [9] that weather turbulences contributes in aircraft accidents with a significant share. Weather could be the only reason for crash, or it can cause pilot errors. According to the air operations workflow, the end-to-end mission preparation process time can take up to 72 hours [16]. Thus, the weather forecast is important in organizing, planning, and executing the daily aerial missions programs.

Thus, the main objective of this paper is to develop a new hybrid simulator for advanced UAVs flight training purposes. The proposed simulator can provide realistic weather conditions for UAV ground pilot; it injects real dynamic weather parameters into the simulation loop, through the exploitation of global weather forecast services as a dynamic weather data source, and by the means of Dryden wind turbulence model.

Towards fulfilling this, the hybrid active architecture (which integrates the VB.NET and MATLAB in the same simulation loop [1]) is used as a development architecture. The proposed simulator relies on Software in-the-Loop (SiL) simulation [17], Discrete Event Simulation (DES) [18], Model in-the-Loop (MiL) simulation [19], human in-the loop simulation [20], and HiL simulation [7]. Moreover, a comprehensive aircraft mathematical model for the De Havilland DHC-2 “Beaver” is utilized to represent the UAV dynamics. Through the COM automation server [21], VB.NET Graphical User Interface (GUI) application is developed to control the whole MATLAB engine processing and to control the exchange commands and feedbacks. In this paper, the implementation of a new weather interface module

Manuscript received July 5, 2015; revised August 25, 2015.

A. Abdelhamid is with College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics (NUAA), 29 Yuda Street, Nanjing 210016, China (e-mail: Amr@nuaa.edu.cn).

P. Zong is with NUAA, Astronautics College, Nanjing 210016, China (e-mail: PengZong@nuaa.edu.cn).

B. Abdelhamid is with Ain Shams University, Faculty of Engineering, Electronics and Communications Department, Cairo, Egypt (e-mail: Bassant.Abelhamid@eng.asu.edu.eg).

is presented to deliver the weather parameters to the proposed simulator. The developed weather module exploits a global weather provider Application Programming Interface (API) [22] to automatically capture the real weather data. Furthermore, based on the real weather data, the Dryden wind turbulence model [23]-[26] is efficiently exploited to generate different linear and angular wind velocities according to the trainee mission parameters such as: altitude, orientation, and speed. The Dryden Simulink model parameters are dynamically changed by VB.NET according to the real weather data during the execution of the simulated mission.

Unlike [1], [2], [4], [12]-[15], [27], the advantages of developing this simulator with the weather forecasting features are as follow: 1) It enhances the capability of the UAV ground pilots to challenge real wind in the simulated environment and achieve the mission goals. 2) It solves the missed reality in the previous UAV simulators by injecting real weather data in the STPs. 3) It provides more capabilities for the mission planner to design safer trajectories. 4) It provides extra visual modalities for UAV ground pilot through the overlaid synthetic weather data layer. 5) It extends the weather data coverage to a global range. 6) It validates the new exploitation of global weather forecast services with no extra budget. 7) It utilizes the active hybrid architecture to implement the MiL (without the need of any auto-code generation or S-function bridge). 8) Its functional fidelity is higher than single environment simulators.

Many difficulties have been conquered in this paper, such as: 1) Integrating all mentioned activities on a single workstation while maintaining the visualization time constraints. 2) Automating the acquisition process of weather data without affecting the training. 3) Handling In-memory database to fetch the corresponding weather zone according to UAV coordinates. 4) Changing Dryden model block parameters dynamically while the simulated mission is running. 5) Achieving high resolution while recording different activities up to 10^{-7} seconds.

The results show that the proposed hybrid simulator achieves realistic training facilities, accurate recording and playback features, advanced dynamic environmental features, and enhanced experience for the trainee against different unpredicted flight situations. Finally, some qualitative aspects of the proposed simulator and different similar simulators are introduced to address its features.

The major contributions of this paper can be summarized as follow:

- 1) To the best of our knowledge, current UAV simulators do not account real weather conditions as a part of UAV STPs. Therefore, an attempt is conducted to consider the real wind turbulence effect as a necessary part of UAV STPs (ranging from mission planning and real operation).
- 2) Design and implement a new hybrid simulator for UAV advanced simulated training using active hybrid architecture. Such simulator enhances the UAV trainee experience against real unpredicted weather situation.
- 3) Introduce a new vision for automating the utilization of global weather forecast services as a mandatory component in the UAV STPs.
- 4) Provide continuous on-line dynamic update of real

weather data during UAV training mission execution.

The rest of this paper is organized as follows: Section II presents the related mathematical preliminaries, Section III details the proposed simulator model with dynamic real weather capability, Section 0 demonstrates the global weather forecast implementation, and Section V illustrates the system integration. Finally, Sections VI and VII comprehensively present the results of the proposed simulator and conclusions, respectively.

II. PRELIMINARIES

The proposed simulator backbone mathematical models are important to provide the UAV dynamics for the trainee. Thus, this section illustrates a brief mathematical background for the main simulator components.

A. UAV Mathematical Model

The proposed simulator utilizes the Flight Dynamics and Control (FDC) toolbox to achieve the UAV dynamics. The FDC toolbox makes it possible to analyze the “Beaver” dynamics and flight control systems within one software environment (MATLAB) on single workstation. The basic Newtonian mechanics are used to derive the aircraft equation of motion, starting with the rigid body force and the moment equations [28]-[30] which are stated as follow:

$$\mathbf{F} = m \left(\frac{\partial \mathbf{V}_A}{\partial t} + \boldsymbol{\Omega}_A \times \mathbf{V}_A \right) \quad (1)$$

$$\mathbf{M} = \frac{\partial (\mathbf{I} \cdot \boldsymbol{\Omega}_A)}{\partial t} + \boldsymbol{\Omega}_A \times (\mathbf{I} \cdot \boldsymbol{\Omega}_A) \quad (2)$$

where m is the aircraft mass, $\mathbf{V}_A = [u \ v \ w]^T$ is the aircraft velocity vector at the center of gravity (c.g.) where u, v, w are the velocity component along the X_{Body} axis, Y_{Body} axis, and Z_{Body} axis, respectively. $\boldsymbol{\Omega}_A = [p \ q \ r]^T$ is the aircraft angular velocity vector about the c.g. where p is the angular rate of roll, q is the angular rate of pitch, and r is the angular rate of yaw, $\mathbf{F} = [F_x \ F_y \ F_z]^T$ is the total external force vector where F_x, F_y, F_z are the total external force along X_{Body} -axis, Y_{Body} -axis, and Z_{Body} -axis, respectively. $\mathbf{M} = [L \ M \ N]^T$ is the total external moment vector where L is total rolling moment, M is the total pitching moment, and N is the total yawing moment, and \mathbf{I} is the inertia tensor of the rigid body. The derivation continues till reaching the external aerodynamic control inputs vector \mathbf{u}_{aero} and output state vector \mathbf{X} as follow [27],[30]:

$$\mathbf{u}_{aero} = [\delta_e \ \delta_a \ \delta_r \ \delta_f]^T \quad (3)$$

$$\mathbf{X} = [V, \alpha, \beta, p, q, r, \psi, \theta, \varphi, x_e, y_e, H]^T \quad (4)$$

where $\delta_e, \delta_a, \delta_r,$ and δ_f are the elevator deflection, aileron deflection, rudder deflection, and flaps deflection, respectively. These inputs are the deflections of aerodynamic control surfaces. The output state vector \mathbf{X} consists of twelve elements: three linear velocities, three angular velocities (p, q, r), three Euler angles which define the attitude of the aircraft relative to the Earth (ψ, θ, φ), two coordinates and the altitude which define the aircraft position (x_e, y_e, H) relative

to the Earth, true airspeed (V), angle of attack (α), and sideslip angle (β).

B. Atmospheric Wind Turbulence Model

Numerical methods are used to generate the linear and angular velocities of atmospheric turbulence to conduct numerical simulations. Dryden model is considered as one of the most attractive models for atmospheric turbulence. Due to the ability of factorizing the Dryden atmospheric turbulence rational formula (which is necessary for the numerical simulations). Thus, this model is utilized in this paper. The Dryden model is a function form that empirically fits the measured power spectral data. It handles the linear and angular velocity components of the continuous gusts as random processes. This model assumes that turbulent gusts are isotropic (the statistical properties do not depend upon the spatial orientation of air mass itself) [23]. The Dryden model is described by Power Spectral Densities (PSDs) for wind gusts three linear velocity components ($u_w, v_w, and w_w$) which can be expressed as follow [24]:

$$\Phi_{u_w}(\Omega) = \frac{2\sigma_{u_w}^2 L_{u_w}}{\pi} \cdot \frac{1}{1 + (L_{u_w}\Omega)^2} \quad (5)$$

$$\Phi_{v_w}(\Omega) = \frac{\sigma_{v_w}^2 L_{v_w}}{\pi} \cdot \frac{1 + 3(L_{v_w}\Omega)^2}{[1 + (L_{v_w}\Omega)^2]^2} \quad (6)$$

$$\Phi_{w_w}(\Omega) = \frac{\sigma_{w_w}^2 L_{w_w}}{\pi} \cdot \frac{1 + 3(L_{w_w}\Omega)^2}{[1 + (L_{w_w}\Omega)^2]^2} \quad (7)$$

where σ_j and L_j are the turbulence intensity and scale length, respectively, for the j^{th} velocity component, and Ω is a spatial frequency (rad./m) [25]. The circular frequency ω can be defined as:

$$\omega = \Omega \cdot V \quad (8)$$

where V is the speed of an UAV flying through a frozen turbulence field. Using the wind gust angular velocity components (p_w, r_w, q_w), the power spectral densities for these components can be expressed as [26]:

$$\Phi_{p_w}(\omega) = \frac{\sigma_{p_w}^2}{VL_{w_w}} \cdot \frac{0.8 \left(\frac{\pi L_{w_w}}{4b}\right)^{1/3}}{1 + \left(\frac{4b\omega}{\pi V}\right)^2} \quad (9)$$

$$\Phi_{r_w}(\omega) = \frac{\mp \left(\frac{\omega}{V}\right)^2}{1 + \left(\frac{3b\omega}{\pi V}\right)^2} \cdot \Phi_{v_w}(\omega) \quad (10)$$

$$\Phi_{q_w}(\omega) = \frac{\pm \left(\frac{\omega}{V}\right)^2}{1 + \left(\frac{4b\omega}{\pi V}\right)^2} \cdot \Phi_{w_w}(\omega) \quad (11)$$

where: $\Phi_i(\Omega) = V\Phi_i\left(\frac{\omega}{V}\right)$, and b represents the UAV wingspan.

III. PROPOSED HYBRID SIMULATOR MODEL

This section focuses on introducing the proposed hybrid simulator model. As shown in Fig. 1, the simulator model integrates multiple techniques to achieve an integrated power rather than relying on a single technique or technology. The main components of the proposed simulator are human-in-the-loop, HiL, SiL, DES, and MiL simulation. The proposed simulator does not require any automatic code generation from MATLAB to any other environment (MiL benefit), does not need network infrastructure (uses single workstation), and deals with dynamic activities (trainee actions and real weather data). From the numbered labels in Fig. 1, it can be observed that the proposed simulator collects the ground pilot guidance commands in steps 1) and 2), captures the real weather forecast data in step 3), then, it sends all parameters to the MATLAB engine in step 4). On the other hand, in step 5), the MATLAB engine receives the simulation parameters from the COM server, in step 6), then, it sends the stochastic wind turbulence parameters to the aircraft mathematical model, in step 7), the mathematical model process guidance commands and the turbulence parameters and sends the results to the GUI. Finally, in step 8), GUI receives the aircraft mathematical model states outputs from the COM client and visualizes them for the ground pilot, and in step 9), all modules and interfaces data are recorded by the data logging module. After the simulated mission is completed, the playback activities are conducted to evaluate the recorded mission through step 10). The following subsections details the simulator workflow and applied methodologies for infrastructure setup.

A. Simulator Software Model

This subsection introduces a brief knowledge about the workflow of UAV simulated mission. During the mission preparation phase, the workflow is as follow: 1) Mission designer determines the interested zones for the required training mission before its due time with 1-3 days [16]. 2) Through GUI capabilities, weather forecasting data are automatically overlaid on the geographic map to support the designer planning decisions. 3) The designed mission trajectory is saved to be ready for the training process. On the other hand, during real training phase, the trainee workflow is as follow: 1) assimilates the mission description and targeted objectives. 2) Loads the assigned mission trajectory. 3) Loads the required weather conditions for designated terrestrial zones. 4) According to mission description, starts the training process. Lastly, at final evaluation phase, the instructor workflow is as follow: 1) reload the data logging files. 2) Run the playback for evaluation/corrective actions. 3) Print signals playback for future analysis and documentation processes.

B. Simulator Setup and Methodologies

Towards fulfilling the mentioned workflow, the proposed simulator utilizes the active hybrid architecture [1]. Using this architecture, VB.NET is utilized for human operator activities as illustrated in Fig. 1. MATLAB/Simulink is responsible of all background mathematical models calculation about UAV dynamics and wind turbulence generation. The proposed simulator utilizes an open-loop "Beaver" mathematical model which is called "ooloop1". The "ooloop1" allows MATLAB users to achieve complete flight parameters due to certain control inputs. "Oloop1" model

solves twelve ODEs (or state equations) and a large number of output equations to obtain all required flight parameters [28]-[31]. Furthermore, via the utilization of COM technology, VB.NET closes “oloop1” simulation loop by exchanging pilot inputs and output state variable to/from the UAV mathematical model. The proposed simulator is developed using single workstation with the following specifications: AMD FX-8350 Black Edition 8-Core 4.0GHz/8MB L2 cache and 8MB L3 cache, 8 GB RAM clocked at 1866MHz, 2GB video card, and equipped with Windows 7 operating system. The proposed simulator

dynamically captures the weather data from global weather provider according to UAV coordinates, and injects them to Dryden model to generate dynamic stochastic atmospheric turbulence components.

These components are integrated with the trainee guidance commands and provided as inputs to the UAV mathematical model. The UAV model is executed to calculate the required output state variables according to the mentioned input combinations. Finally, the data logging module is developed to record whole system input and output components for further analysis procedures.

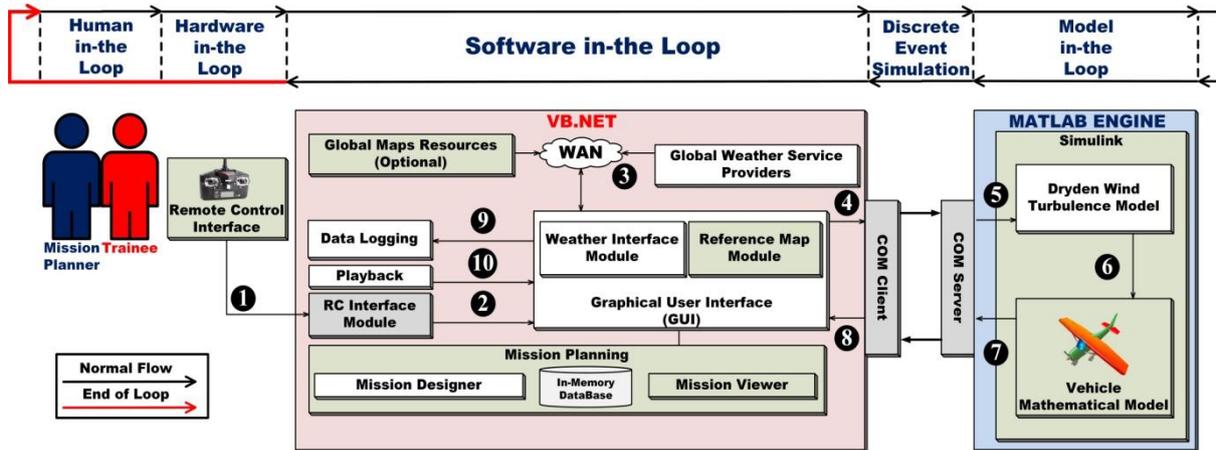


Fig. 1. Proposed hybrid simulator block diagram with data flow sequence.

IV. REAL WEATHER FORECAST INTEGRATION

This section introduces a global weather forecast service sample (to validate this new approach for UAV simulators). Additionally, the developed software interface module is demonstrated to explain the automation process of providing continuous real weather data for the proposed simulator.

A. Global Weather Provider API

Typically, most of the time, weather data frame contains the following parameters: observation time, temperature, wind direction, wind speed, humidity, visibility, and pressure. Therefore, by selecting an appropriate weather provider (local country or worldwide), the weather forecast data could be achieved. Obviously, this could be done after accomplishing some registration and interfacing procedures. Most of worldwide weather providers use APIs for supplying weather data to customers. The proposed hybrid simulator exploits “WorldWeatherOnline” as a worldwide instantaneous weather provider (as a sample to validate the concept). The weather provider online API explains the structure and formats of this data exchange service.

B. Weather Interface Module Development

For achieving more realistic simulated training, it is required to implement a new software module to be responsible about the weather provider requests/responses. This module is important due to its role in both mission planning and operational training processes. According to the mission objective, the required weather parameters could have the current weather data or the forecasting one. As a result, to get maximum benefits from the above mentioned

service, an automated process is developed for both types of current and forecast weather data. Fig. 2. demonstrates the flowchart of the weather interface module, which is implemented as a standalone thread. This module is responsible of the web communication with the weather providers through the implementation of the provider’s APIs. Furthermore, this module parses the provider responses into useful information and stores them into the in-memory database. These activities are controlled from mission planning tab (as will be shown in the results section).

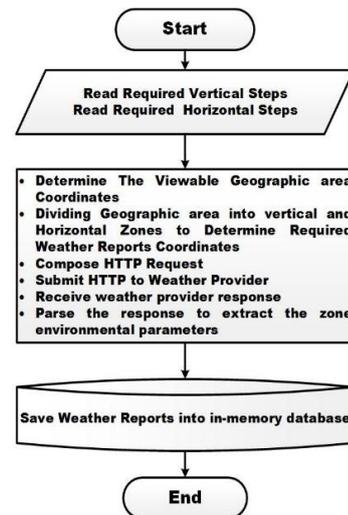


Fig. 2. Weather interface module flowchart.

V. INTEGRATION OF UAV MATHEMATICAL MODEL WITH DRYDEN ATMOSPHERIC MODEL

Using the active hybrid architecture power, the backbone

VI. SIMULATION RESULTS

As the Software Development Life Cycle (SDLC) procedures state that checkpoints and testing procedures are mandatory after module development [1], [32], a performance analysis checkpoint is implemented to test the integration of all developed modules. This checkpoint contains many test procedures to evaluate the following: 1) proposed simulator capabilities and its visualization time constraints. 2) Effect of integrating weather data on different simulated missions. 3) Signals playback for UAV trainee challenging realistic wind in a simulated mission. 4) Some qualitative aspect of the proposed hybrid simulator and other similar existing ones.

Algorithm 1 Dynamic algorithm for updating Dryden model with real data.

```

1 : initialize
2 : Submit weather requests for the GUI viewable area
3 : Store the weather provider response into in-memory database
4 : loop
5 :   Pull UAV coordinates from MATLAB
6 :   Set turbulence = on
7 :   if UAV position inside known weather zones then
8 :     Search for appropriate weather zone according to the UAV
       coordinates
9 :     Fetch the stored weather detailed parameters
10:    Update Dryden block masked parameters (wind speed, wind
      direction)
11:  else
12:    Submit weather requests for the required weather zones
13:    Store the weather provider response into in-memory database
14:    Update Dryden block masked parameters (wind speed, wind
      direction)
15:  end if
16: end loop
    
```

A. Proposed Software Capabilities and Visualization Time Constraints

As shown in Fig. 4, the proposed hybrid simulator has an innovative GUI that can simultaneously display multiple static layers (map, trajectory, waypoints... etc.) and dynamic ones (UAV symbol, gauges... etc.). The demonstrated mission starts at China, passes over Thailand, and its final destination is Dubai. The trainee loads the trajectory and requests to display the weather forecast data for the next two advanced days. The trainee issues the guidance commands to overcome the wind turbulence along the whole trajectory.

At speed *87 Knots*, the parameters of the injected real weather fluctuate the attitude gauges *5000 ft* and let the trainee have the same feeling of the cockpit. It is worthily noted that the proposed simulator has some advanced features to enhance the simulation process such as: (1) magnifying real weather data with any custom factor for extra training difficulties. (2) Presenting advanced gauges such as vertical speed, altimeter, attitude indicator, heading, and air speed. (3) Providing the recording and playback facilities. These features increase the functional fidelity [3] of the proposed simulator.

Furthermore, Fig. 5 (a) demonstrates the capability of the developed weather interface modules which can retrieve the weather reports for any global coordinate, while, in Fig. 5 (b), a generated color map is illustrated to represent different longitudinal and latitudinal zones. The vertical color bar represents the wind speed in m/s. The color zones map could

be exploited to support the off-line manual trajectory planning and automatic safe path generation. In another words, the blue color from any source coordinate to another destination coordinate represents a trajectory with minimum wind turbulence (safe trajectory). Finally, it is clear that the weather integration to the flight simulators plays an active role during both design and training phases.

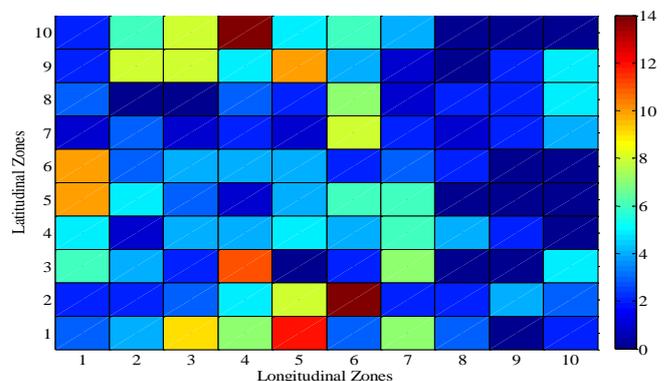
According to the system constraints, the performance of the proposed simulator should satisfy the visualization constraints (30 frames/second), which means that whole simulation loop including the weather simulation should be accomplished in less than 33ms. Therefore, the developed data logging module can be used for accurate measurements of different simulator modules. Therefore, it is used to measure the consumed time to/from the MATLAB COM automation server which represents the heavy backbone computation engine.

As shown in Fig. 6, the resolution of the data logging module is very high, the units of the measurements is in ticks (where $1 \text{ tick} = 10^{-7} \text{ second}$), such measurements accuracy is considered one of the conquered challenges in this paper.

Through this figure, 10000 samples of measurements are plotted; these measurements represent the MATLAB COM automation server commands execution times (processing delay). The actual processing delay variations are shown in blue, while the average MATLAB COM automation server processing delay is shown in red color. It can be deduced that, using the active hybrid architecture technique, the average delay is in order of 9.54ms, which means that the proposed simulator architecture is below 33ms time constraints. As a result, a graphical smooth motion of graphical features is achieved.



(a)



(b)

Fig. 5. Global weather forecast data (a) synthetic weather layer, (b) generated color map representation.

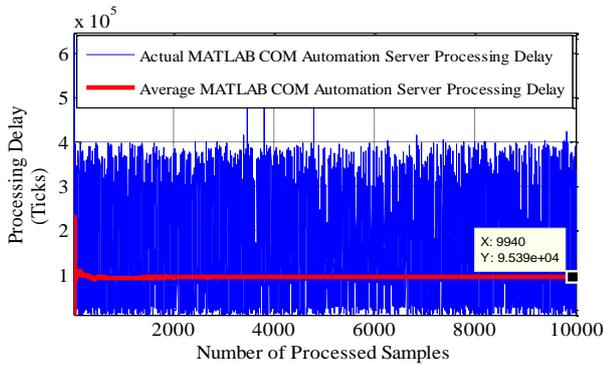


Fig. 6. MATLAB/Simulink COM automation server processing delay with reference to the platform.

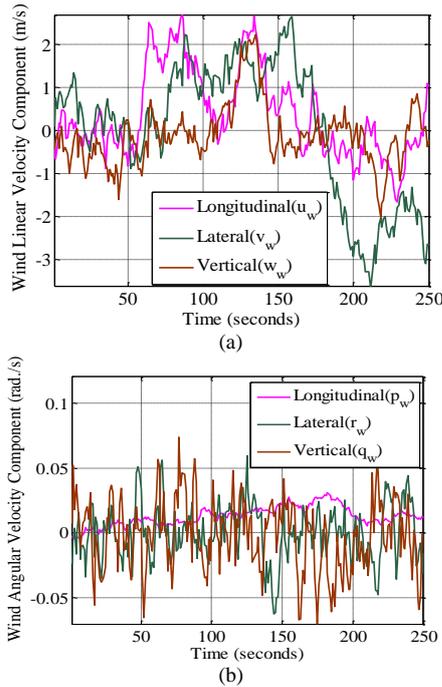


Fig. 7. Stochastic wind velocities based on real weather data (a) linear components, (b) angular components.

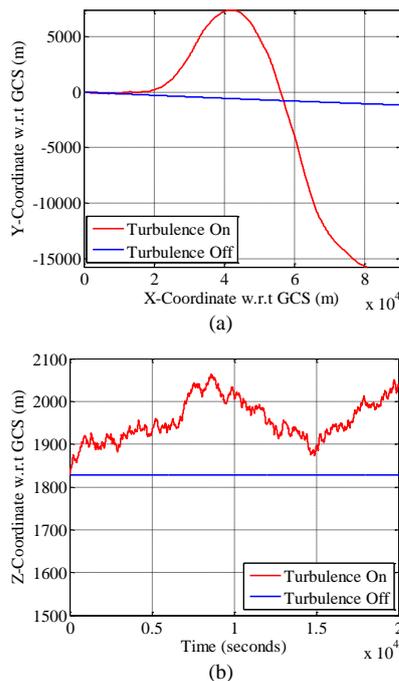


Fig. 8. Simulation results comparison for altitude hold cruise missions based on Dryden's model (a) XY plan, (b) Z plan.

B. Real Weather Reports Effect on Different Simulated Missions

Through this subsection, a unified concept has been established for better understanding of the figures. This mission is conducted to present the capability of the proposed simulator to inject real weather reports into the simulation process, and generated realistic turbulence components that interactively affect the UAV during the flight.

As shown in Fig. 7, a real weather report (north wind with 6 m/s) is converted to a complete set of linear and angular wind velocities corresponding to the UAV altitude. It is clear that, at 6000 ft altitude, the wind linear velocity components randomly vary between -3.5 to $+2.8$ m/s, while the angular velocities components stochastically vary between -0.07 to $+0.07$ rad/s.

Additionally, Fig. 8 represents a cruise mission with altitude hold at 1828m (6000 ft.). The human operator performs two missions without any guidance commands at all. One ideal mission is conducted without any wind parameters, while the second mission is conducted with an existed wind. For the turbulence off mission, as shown in Fig. 8 (a), the De Havilland DHC-2 "Beaver" slightly deviates to the left due to propeller moment along the X-axis, which appears in the aircraft mathematical model as negative stability coefficient C_{y_0} in [27]. For turbulence on scenario, the stochastic wind components let the UAV wind deviates to the right, while, after 40 km, stochastic wind components are changed in the opposite direction and lets the UAV deviated to the left with reference to the idea mission trajectory.

Moreover, due the stochastic generation of the vertical wind component, Fig. 8 (b) shows that UAV altitude is no longer maintained at 1828m (6000 ft.). Such vertical wind component lets the altitude fluctuate randomly above the ideal case (6000ft.). This result ensures the generation of realistic 3D wind turbulence by the proposed simulator model.

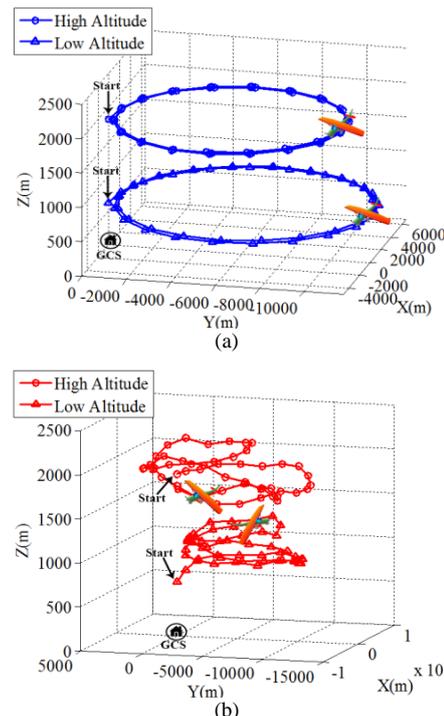


Fig. 9. Instability behavior for different altitude circular missions. (a) 3D mission with turbulence off (b) 3D mission with turbulence on.

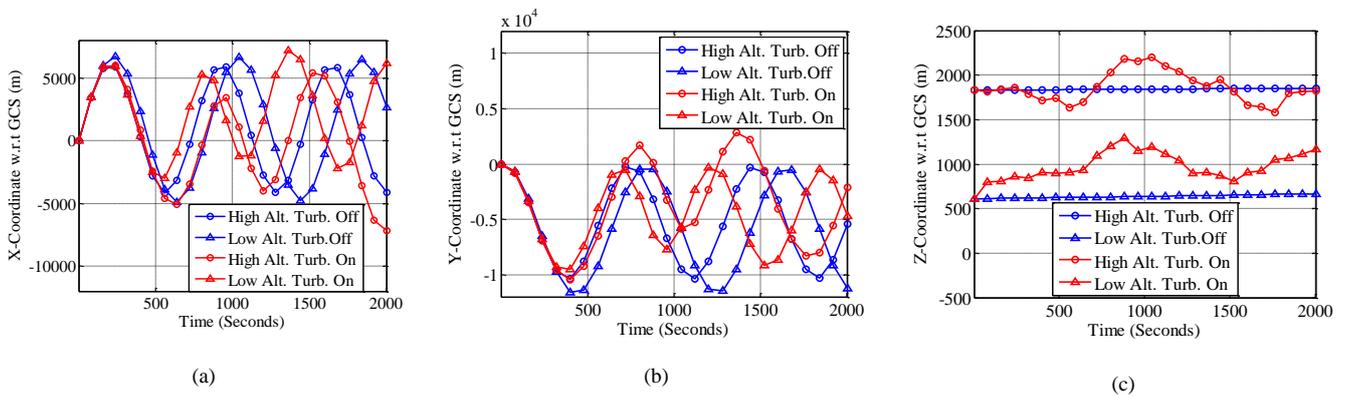


Fig. 10. Different profiles for different altitudes circular missions. (a) X profile, (b) Y profile, and (c) Z profile.

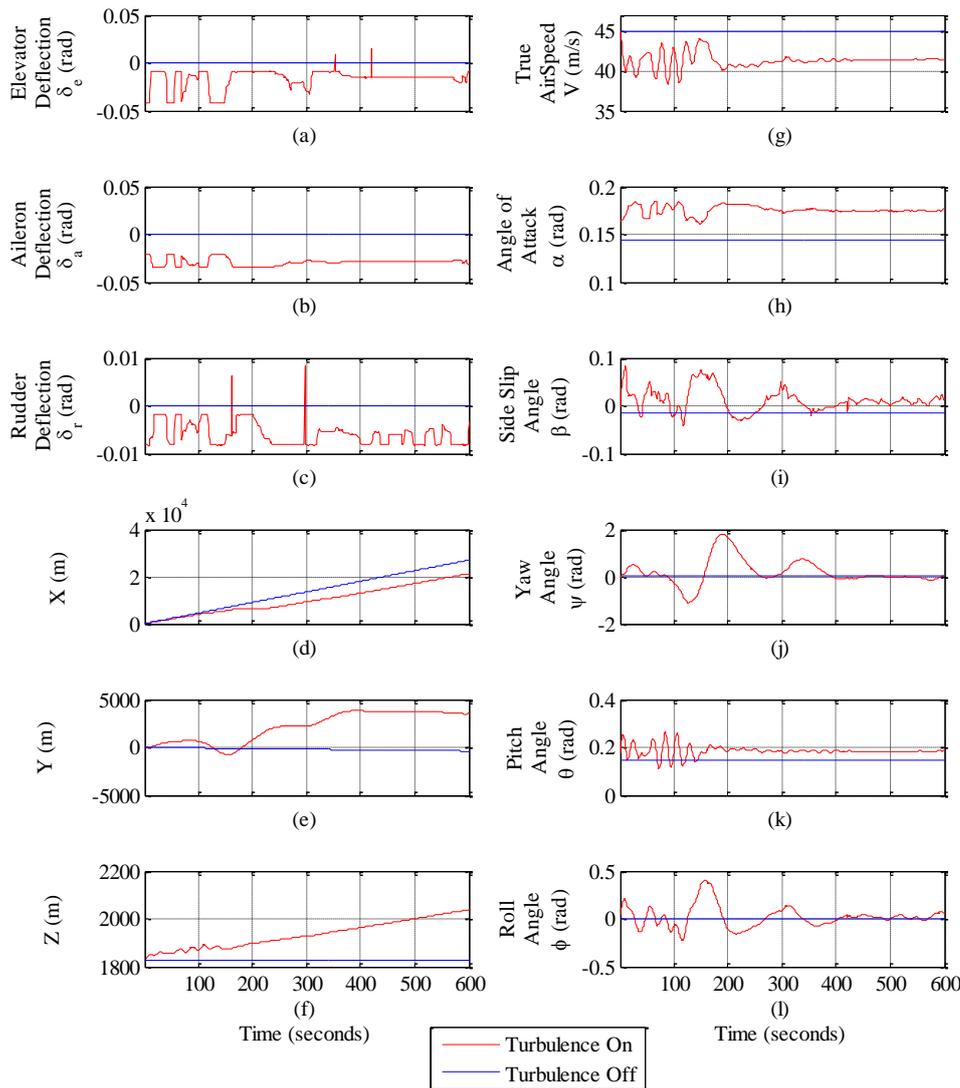


Fig. 11. Trainee interactive mission profile to combat wind turbulence and maintain certain heading with reference to ideal mission without wind.

Finally, Fig. 9 demonstrates two circular missions conducted on different altitudes 610 m (2000 ft.) and 1828 m (6000 ft.). These missions are conducted at the same geographical zone to analyze the effect of the injected real weather on the UAV subject to its altitude. Fig. 9 (a) shows 3D representation of the two ideal missions without any injected wind parameters, while Fig. 9 (b) demonstrates 3D representation of the wind effect on both different altitude missions. It is clear that the wind turbulence affects the circular trajectory and the mission height, and these results

validate the integration between the UAV mathematical model and the Dryden turbulence model. The UAV starts to deviate to the positive y-coordinates in existence of wind, instead of complete negative coordinates in the ideal mission (no wind). Fig. 10 represents the individual axes behavior of the conducted circular missions with respect to GCS. It can be deduced that the proposed simulator can generate different 3D wind turbulence components for different flying altitudes, which imitates the real wind characteristics in [25]. From this experiment, trainee can realize the realistic 3D wind effects

during the simulated training process, which provides extra mission awareness and enhance the trainee capabilities.

TABLE I: PROPOSED HYBRID SIMULATOR VS. OTHER SIMULATORS

Criteria	Proposed Advanced Hybrid Simulator	Active Hybrid Architecture Simulator [1]	IEST Fixed Based Flight Simulator [12]	Low Cost Virtual Simulator [15]	Sejong UAV FlightSim (SUF) [14]
Flight Model	Fixed Wing De-Havilland DHC-2 "Beaver"	Fixed Wing De-Havilland DHC-2 "Beaver"	IEST framework internal flight model	Helicopter Uh-1h	Fixed Wing De-Havilland DHC-2 "Beaver"
Computing Platform	1 Pc	1 Pc	10 Pc	3 Pc	1 Pc + cPCI Pc
OS	Windows	Windows	Windows + QNX	Windows	Windows + QNX
Atmospheric Models	Dryden Wind Turbulence Model	Use FDC Models	Not explicitly Mentioned	Not Available	Use FDC Models
Atmospheric Model Flexibility	Flexible architecture to replace Dryden with any other model.	Not Available	Not explicitly Mentioned	Not Available	Select one of the FDC models
Real Wind Visualization in Training Missions	Available	Not Available	Not Available	Not Available	Not Available
Mission Planning with Real Weather Support	Available	Not Available	Not Available	Not Available	Not Available
Real Weather Forecast	Available	Not Available	Not Available	Not Available	Not Available
Real Weather Data Recording Module	Available	Not Available	Not Available	Not Available	Not Available
Signals Playback for Real Wind parameters	Available	Not Available	Not Available	Not Available	Not Available
Human Interactive missions for Real Wind Challenging	Available	Not Available	Not Available	Not Available	Not Available
Wind Magnification	Available	Not Available	Not Available	Not Available	Not Available
GUI	Available (VB.NET)	Available (VB.NET)	Available	Available	Visual Basic/Open GL
Functional Fidelity	Above medium	Above medium	Not explicitly Mentioned	Medium	Above Medium

C. Signals Playback for an Interactive Human in the Loop Mission (Wind Challenging Mission)

Fig. 11 illustrates the signals playback for a non-novice trainee who performs a cruise mission in existence of wind turbulence. The main mission objective is to maintain zero heading ($yaw\ angle\ (\psi) = 0^\circ$). The mission is flying over a geographical zone ($30\ Km^2$) has the following real weather conditions (wind speed= $6m/s$ and wind direction= 73°). The ideal mission with zero wind is illustrated in blue color. This mission demonstrates the whole interactive training process illustrated in Fig. 11 (the trainee activities, the real weather conditions, the stochastic generation of linear and angular wind components, and the resultant UAV response).

As shown in Fig. 11 (a), (b), and (c), to overcome realistic wind turbulence effect, the trainee issues the required compensating guidance commands to the elevator (δ_e), aileron (δ_a), and rudder (δ_r) control surfaces, which represent the external aerodynamic control inputs (\mathbf{u}_{aero}) of the mathematical model. Spikes in these figures are premeditated to show the rapid data acquisition of software modules. These aerodynamic inputs in addition to real wind parameters are injected to the mathematical model to calculate the new UAV output state variables illustrated in Fig. 11 (d) to (l). Fig. 11 (d), (e), and (f) present X , Y , Z coordinates, respectively, which reflect the turbulence effect on the UAV trajectory such as: gaining extra altitude and

swaying to the right and left due to lateral wind components. Fig. 11 (g), (h), and (i) represent the wind effect on UAV speed, angle of attack, and side slip angle, respectively. It is also clear that the trainee compensation commands in addition with the wind components affect the speed and the other factors with respect to the ideal mission. Focusing on Fig. 11 (j), the trainee fulfills the mission objective by maintaining the UAV heading around the demanded value ($\psi) = 0$. It is clear that the trainee consumes the first 300 seconds to figure out the required compensation signals to combat such wind. Conversely, during the last 300 seconds, the trainee achieves the mission goal by adjusting the UAV heading (ψ) around zero degree.

Fig. 11 (k) and (l) present the effect of the trainee compensation commands to stabilize the roll and pitch angles around their stable values. It can be deduced from this mission that rudder actuator is utilized intensively to maintain zero heading objective, which can be translated to a recommendation for adjusting the replacement strategy of critical UAV components. The trainee achieves high functional fidelity about the wind effects from the simulator gauges, and challenges the wind efficiently to achieve the mission goal.

D. Qualitative Aspects of the Proposed Hybrid Simulator vs. Different Existing Simulators

Table I lists some qualitative aspects of the proposed

hybrid simulator versus different other simulators. It is clear that the proposed simulator has advanced features by providing real weather conditions for the human operator during the mission planning and real training phases. Moreover, the training process achieves more reality, and its cost still feasible to be implemented by other researchers.

VII. CONCLUSION

This paper presented a new hybrid simulator with real weather forecasting capabilities for advanced UAV flight training. The proposed simulator hybrid architecture consists of many components and utilizes many advanced technologies in the same simulation loop. It utilizes a comprehensive MATLAB/Simulink aircraft mathematical model (to achieve the UAV dynamics), and a VB.NET developed software for different training activities. The integration of the VB.NET with the MATLAB/Simulink increases the functional fidelity of the proposed simulator. This paper demonstrated a new technique for exploiting global weather forecasting reports for UAV flight simulators. The results show that the proposed simulator visualization requirement is achieved, the exploitation of real global weather forecast data for UAV flight simulators is validated, different stochastic wind generations for different mission altitudes are approved, and the advanced training of the ground pilots in existence of wind turbulence is accomplished. It is observed that, with an inclusion of weather turbulence effects, the trainee's performance can be significantly improved to achieve mission success within short period of time. Additionally, it can train the operator for different predicted and unpredicted situations. Furthermore, it is inferred from the control surfaces profile that providing rudder compensation is significant to accomplish a fixed heading cruise mission in existence of wind turbulence. Finally, some qualitative aspects of the proposed simulator and other different types of simulators were listed to address the proposed features.

REFERENCES

- [1] A. A. Hamid and P. Zong, "A novel software simulator model based on active hybrid architecture," *International Journal of Aerospace Engineering*, p. 19, 2015.
- [2] A. A. Hamid and P. Zong, "Software design flexibility based on dual environment architecture: Aerial case study," in *Proc. International Conference on Computer Information Systems and Industrial Applications*, 2015, pp. 146-149, Bangkok.
- [3] J. Craighead, R. Murphy, J. Burke, and B. Goldiez, "A survey of commercial & open source unmanned vehicle simulators," in *Proc. IEEE International Conference on Robotics and Automation*, pp. 852-857, Roma, 2007.
- [4] A. Ajami, J. Balmat, J. P. Gauthier, and T. Maillot, "Path planning and ground control station simulator for UAV," in *Proc. 2013 IEEE Aerospace Conference*, 2013, pp. 1-13, Montana.
- [5] W. Jianan, W. Wei, Z. Jinhong, and W. Bodong, "Research of a kind of new UAV training simulator based on equipment simulation," in *Proc. 2011 International Conference on Electronic and Mechanical Engineering and Information Technology (EMEIT)*, 2011, pp. 4812-4815, vol. 9, Harbin.
- [6] A. H. Swaroop, T. Bharat, and A. Hemendra, "Distributed hardware-in-loop simulations for multiple autonomous aerial vehicles," in *Proc. AIAA Modeling and Simulation Technologies Conference*, American Institute of Aeronautics and Astronautics, 2015.
- [7] M. Rashmi and A. Hemendra, "MAV - Hardware In Loop Simulation using LabVIEW," *AIAA Modeling and Simulation Technologies Conference*, American Institute of Aeronautics and Astronautics, 2011.
- [8] L. Mariano, D. Vladimir, E. Gabriel, C. Renwick, and K. Isaac, "Simulink based hardware-in-the-loop simulator for rapid prototyping of UAV control algorithms," in *Proc. AIAA Infotech@Aerospace Conference*, American Institute of Aeronautics and Astronautics, 2009.
- [9] P. C. I. Website. (2015). Accident statistics. [Online]. Available: <http://www.planecrashinfo.com/cause.htm>
- [10] R. E. Weibel and R. J. Hansman, *Safety Considerations for Operation of Unmanned Aerial Vehicles in the National Airspace System*, MIT Press, 2006.
- [11] J. Shmelev, "Simulator training optimization of UAV external pilots," in *Proc. 2014 IEEE 3rd International Conference on Methods and Systems of Navigation and Motion Control (MSNMC)*, 2014, pp. 75-78, Kyiv.
- [12] S. Zheng, Z. Ye, J. Jin, and J. Han, "Fixed-based flight simulator development for Human-machine Interaction," in *Proc. International Conference On Computer and Communication Technologies in Agriculture Engineering (CCTAE)*, 2010, pp. 408-411, vol. 2, Chengdu.
- [13] S. Zheng, Q. Huang, J. Jin, and J. Han, "Flight Simulator Architecture Development and Implementation," in *Proc. 2009. ICMTMA '09. International Conference on Measuring Technology and Mechatronics Automation*, 2009, pp. 230-233, vol. 2, Hunan.
- [14] H. Cheolkeun, Y. Steve, and H. Myoung-Shin, "Development of a UAV simulator based on COTS products and object-oriented programming languages," in *Proc. AIAA Modeling and Simulation Technologies Conference and Exhibit*, American Institute of Aeronautics and Astronautics, 2001.
- [15] I. Yavrucuk, E. Kubali, and O. Tarimci, "A low cost flight simulator using virtual reality tools," *IEEE Aerospace and Electronic Systems Magazine*, vol. 26, no. 4, pp. 10-14, 2011.
- [16] J. M. Wohletz, D. A. Castanon, and M. L. Curry, "Closed-loop control for joint air operations," in *Proc. 2001 American Control Conference proceedings*, 2001, pp. 4699-4704, vol. 6, Arlington, VA.
- [17] C. Bonivento, M. Cacciari, A. Paoli, and M. Sartini, "Rapid prototyping of automated manufacturing systems by software-in-the-loop simulation," in *Proc. 2011 Chinese Control and Decision Conference (CCDC)*, 2011, pp. 3968-3973, Mianyang.
- [18] L. F. Bertuccelli and M. L. Cummings, "Operator choice modeling for collaborative UAV visual search tasks," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 42, no. 5, pp. 1088-1099, 2012.
- [19] V. Jalili-Marandi, J. Belanger, and F. J. Ayres, "Model-in-the-loop real-time simulation in phasor domain," in *Proc. 2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, 2014, pp. 2280-2284, Istanbul.
- [20] D. Nunes, P. Zhang, and J. Silva, "A survey on human-in-the-loop applications towards an Internet of all," *IEEE Communications Surveys & Tutorials*, no. 99, pp. 1-10, 2015.
- [21] L. Manmaker, "MATLAB external interfaces," *The MathWorks*, 2013.
- [22] L. Jun, X. Yingfei, L. Xuanzhe, and Z. Lu, "How does web service API evolution affect clients?," in *Proc. 2013 IEEE 20th International Conference on Web Services (ICWS)*, 2013, pp. 300-307, Santa Clara, CA.
- [23] E. F. Hogge, "B-737 linear autoland simulink model," NASA, Langley Research Center, Hampton, Virginia, USA, Tech. Rep. CR-2004-213021, 2004.
- [24] S. Fortunati, A. Farina, F. Gini, A. Graziano, M. S. Greco, and S. Giompapa, "Impact of flight disturbances on airborne radar tracking," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 48, No. 3, pp. 2698-2710, 2012.
- [25] D. Moorhouse and R. Woodcock, "US military specification MIL-F-8785C," Technical Report, US Department of Defense, 1980.
- [26] The MathWorks, *Dryden Wind Turbulence Model*, MATLAB Reference Pages, 2010.
- [27] M. Rauw, *FDC 1.2 - A Simulink Toolbox for Flight Dynamics and Control Analysis*, Delft, The Netherlands, 2001.
- [28] E. L. Duke, R. F. Antoniewicz, and K. D. Krambeer, *Derivation and Definition of a Linear Aircraft Model*, NASA Reference Publication 1207, 1988.
- [29] B. Etkin and L. D. Reid, *Dynamics of Flight: Stability and Control*, Wiley, New York, 1996.
- [30] R. T. H. Tjee and J. A. Mulder, *Stability and Control Derivatives of the De Havilland DHC-2 "Beaver" Aircraft*, Delft University of Technology, Faculty of Aerospace Engineering, Section Stability and Control, 1988.
- [31] D. McLean, *Automatic Flight Control Systems*, Prentice Hall, 1990.
- [32] H. Hashimi, A. Hafez, and M. Beraka, "A novel view of risk management in software development life cycle," in *Proc. 12th International Symposium on Pervasive Systems, Algorithms and Networks (ISPAN)*, 2012, pp. 128-134, San Marcos, Texas.



Amr Abdelhamid received his B.Sc. in computer engineering, with grade very good with honor, Cairo, Egypt, in 1997; junior software programmer until 1998, senior software programmer until 2003. In 2006, he has received the M.Sc. degree in electronics and communications engineering, in autonomous UAV fuzzy logic controller, Cairo University, Cairo, Egypt. Following qualification is in system quality assurance in 2007. He is interested in data protocols and interfaces, real time systems, and mostly concerned in mobile communications. He worked as a project manager in 2008; and as a principal manager in 2012. Currently, he is a Ph.D. candidate in Nanjing University of Aeronautics and Astronautics (NUAA) at College of Electronic and Information Engineering, Nanjing, Jiangsu, China.



Bassant Abdelhamid received her B.Sc. and M.Sc. degrees in communications and electronic engineering from Ain Shams University, Egypt in 2006 and 2010, respectively, and received her Ph.D. in communications and electronic engineering from E-JUST University in 2013. She worked at Ain Shams University as a demonstrator, teaching assistant, and assistant professor in 2006, 2010, and 2013, respectively. During her Ph.D., she worked as a special research student at Kyushu University, Japan. She is specialized in communications engineering, digital signal processing, and mobile communications.



Peng Zong received the B.Sc. in Nanjing University of Aeronautics and Astronautics (NUAA), China and Ph.D. degree from University of Portsmouth United Kingdom and has been a professor in College of Astronautics NUAA since 2006. His academic experience is as a research fellow in Centre for Communication Systems Research, University of Surry, England in 2001, following by Employee of Hughes Network System (USA) as a software engineer and a chief designer in Information and Technology Institute of "China Aerospace Science & Industry Corp" (CASIC) until 2006. His research fields are mobile communication, satellite networking mostly concerned by routing algorithm of LEO constellation, and stealthy detection of netting radar.