An AI Knowledge Model for Self-Organizing Conflict Prevention/Resolution in Close Free-Flight Air Space[1]

Ivan Y. Burdun

[obsolete affiliation information omitted]

Oleg M. Parfentyev

[obsolete affiliation information omitted]

Abstract—An artificial intelligence (AI) knowledge model is proposed as a basis for automated conflict management in close free-flight air space. A conceptual framework of the model is developed. The problem of air traffic control (ATC) within a group of potentially conflicting vehicles is represented as an autonomous, self-organizing process. This process incorporates the principles of collective behavior observed in nature (bird flocking, fish schooling, insect swarming, etc.) and comprehensive knowledge of the "pilot - vehicle - operational environment" system dynamics derived from special computer experiments. Through this process, real-time knowledge-based predictions of near term flight paths are made to prevent and resolve conflicts in a group of vehicles under normal and demanding conditions. The model may be useful for prototyping intelligent technologies for assisted and automatic collision avoidance in close air space.

TABLE OF CONTENTS

1. INTRODUCTION

Definition—Free Flight (FF) is *"a safe and efficient flight operating capability under instrument flight rules (IFR) in which the operators have the freedom to select their path and speed in real time. ... Air traffic restrictions are only imposed to ensure separation ... and ... flight safety. ... Free flight in its mature state is intended to provide aviation users visual flight rules (VFR) flexibility while maintaining the traditional protection afforded under IFR by using advanced technology. Intervention is limited to ... tactical (short term) conflict resolution ... and safety of flight"* [1]. Therefore, future FF technologies must be capable of monitoring, prediction and resolution of collision threats in close air space.

External vs. internal control—There may be difficulties in controlling air traffic in a close free-flight situation (FFS) [2] from the ground. Alternative methods for coherent navigation of a group of vehicles in such situations should be explored. One of possible solution approaches is to substitute external control by self-organizing, knowledge-based behavior within a

[1] *Proc. Of 1999 IEEE Aerospace Applications Conference, Snowmass, Colorado, March 6-13, 1999, USA*, Vol. 2, IEEE, 1999, pp. 409-428.

group of potentially conflicting vehicles. To implement such 'internal' control it is proposed to combine several complementary methods. These include the principles of collective behavior observed in nature (bird flocking, fish schooling, insect swarming and animal herding), mathematical modeling and computer simulation of the "pilot - vehicle - operational environment" system behavior, and artificial intelligence techniques. The overall objective is to maintain safe and efficient air traffic flow in a crowded FF environment. However, this task requires comprehensive predictive knowledge of complex system dynamics under multi-factor conditions.

Paper scope and objective—In this paper, an attempt is made to draft a generic conceptual framework of an AI knowledge model for conflict prevention and resolution in close FF air space. The objective is to demonstrate how several advanced methods can be combined into a single generic model for automated conflict management.

2. PROBLEM

Future air traffic—It is anticipated that the volume of global air transport operations will increase dramatically during the next ten-twenty years [3]. As a result, some local air spaces may become overcrowded similar to the current traffic situations observed in large cities. Overcrowding of local air space under FF rules may have even worse consequences for a number of reasons. First, air traffic proceeds in three-dimensional space, and aircraft have six degrees of freedom of motion versus three degrees in cars. Aircraft fly much faster and have a bigger inertia compared with cars. Aerodynamic properties of a modern aircraft are essentially non-linear and can be non-stationary, which make its flight path less predictable for the pilot and avionics. Also, most aircraft cannot decelerate quickly enough or stop in the skies for the sake of traffic control.

Requirements—"Some future FF scenarios envisage considerably greater flight path unpredictability than is typical currently in ATC" [4]. Indeed, a FFS may involve a group of several vehicles merging arbitrarily in close air space. Future FF technologies must ensure safe separation between participating vehicles and satisfy individual goals and constraints. Also, these technologies must be capable of accounting for various demanding operational conditions (factors). The latter include (Fig. 1): pilot's decisions and errors, onboard hardware malfunctions and software errors, adverse weather, and combinations of these factors. As the result, a potential for collision(s) may suddenly develop in a FFS. Therefore, methods for real-time monitoring, prediction, evaluation and correction of individual flight paths within a group of vehicles will be required to support FF. Obviously, these methods should be based on comprehensive knowledge of the group dynamics under anticipated flight conditions.

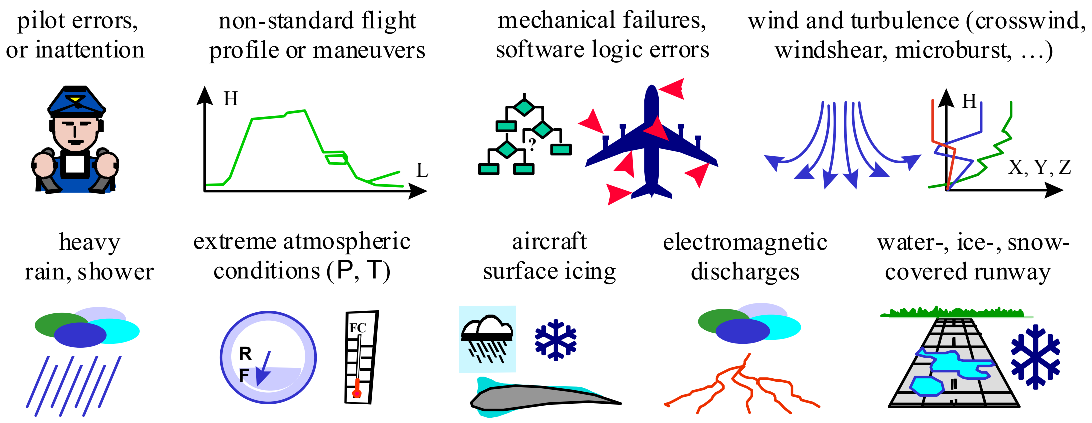


Figure 1: Main groups of anticipated operational conditions (factors) of flight

Problem formulation—Let a group of N heterogeneous vehicles, $N = f(t)$, merge arbitrarily in a close air space of arbitrary shape, which has a characteristic dimension of 4-6 km. Also, flight of these vehicles at any time may be affected by various demanding factors (see Fig. 1). The problem is how to organize the group behavior in order to avoid collisions under normal and demanding conditions. As a part of this formulation, the following task is addressed in the paper: how to formalize, generate and use onboard knowledge of a complex FFS domain to achieve this goal.

3. SOLUTION APPROACH

Role of structure—The following quotation probably best characterizes the suggested solution approach. *"After all, complicated tasks usually do inherently require complex algorithms, and this implies a myriad of details. And the details are the jungle in which the devil hides. The only salvation lies in structure"* [14]. Indeed, the role of correct methodological structures, or knowledge models, in complex systems analyses is vital.

Component methods—To address the problem, an integrated approach is proposed. The approach combines the following methods:

- the principles of self-organizing behavior observed in nature (in bird flocks, fish schools, insect swarms [5])
- a generic mathematical model of the "pilot - vehicle - operational environment" system behavior in complex flight situations [6]
- autonomous simulation experiments with the model for examining complex flight situations [7]
- a fuzzy situational tree-network of flight as an onboard knowledge base of a complex FFS domain [8], [9]
- intelligent formats for conveying this knowledge to the pilot [10].

Purpose—The purpose of combining these methods is two fold: (1) to self-organize behavior within a group of vehicles to avoid potential conflicts in close air space, and (2) to resolve impending conflicts if a demanding situation suddenly develops.

4. INTRODUCTION TO COMPONENT METHODS

In this section, the background and role of each of these component methods is briefly reviewed.

Natural principles of self-organizing collective behavior—Reynolds [5] has developed computer models, which can emulate collective behavioral patterns, which natural species apply so successfully to maintain safe separation in a group motion. Classical examples are bird flocking, fish schooling, insect swarming, and animal herding. Reynolds calls these patterns 'simple steering behaviors' [5], [11]-[13]. These patterns include Seek and Flee, Pursue and Evade, Wander, Arrival, Obstacle Avoidance, Containment, Wall Following, Path Following, and Flow Field Following. They may be used as building blocks in bird navigation on a higher level. It is expected that similar principles can be applied to the FF problem to help avoid potential conflicts in congested air spaces. Note that Reynolds' models require only geometric and kinematic input parameters of flight of individual vehicles. A comprehensive survey of research into computer modeling of coordinated motion can be found in [11], [12]. Online simulations of the simple steering behaviors developed by Reynolds are demonstrated at [13].

Mathematical modeling of the "pilot - vehicle - operational environment" system—In addition to self-organizing behavioral principles based on kinematic constraints, quantitative knowledge of the "pilot - vehicle - operational environment" system dynamics is required to predict vehicle's flight paths under complex conditions. A generic situational model of the 'pilot – vehicle - operational environment' system will be employed as a knowledge generator. This model is an established flight analysis tool developed for studying complex operational situations [6], [7]. Basically, formal objects of three types are required to construct a comprehensive model of a complex flight situation. These are the flight event, the flight process, and the flight scenario. The flight scenario is a plan for implementing a flight situation and associated piloting tactics in simulation or operation.

Autonomous flight simulation experiments—Using flight scenarios and the situational model of flight, systematic knowledge of a non-standard FFS domain can be obtained in computer experiments [7], [15]. These experiments are autonomous because pilot's actions and other operational factors (see Fig. 1) are modeled on a computer together with the vehicle flight dynamics. Other distinguishing features of this method include:

- representation of pilot's tactics and operational factors on the level of cause-and-effect relationships
- accurate mapping of all details of a real or hypothetical situation into flight scenarios
- easy planning and execution of simulation experiments by a non-pilot and non-programmer
- faster-than-real-time flight simulation speed (~40 times).

Fuzzy situational tree-network of flight—To retain and access this knowledge onboard, the concept of fuzzy situational tree-network (FSTN) is employed – ref. [6], [8], and [9]. Basically, the FSTN is an artificial memory structure of extremely large volume (~10^1-10^3 Gbytes) and open architecture. It is constructed and stored as a tree of many interrelated fuzzy flight path-branches, which are specially planned to thread a complex FFS domain. Its purpose is to explore critical combinations of key operational factors of flight and their possible adverse effects (see Fig. 1). Note that this knowledge cannot be obtained from other sources.

In a free flight situation the FSTN is used for near-term predictions of flight paths for a vehicle under non-standard conditions. This process is automatically activated when an initially safe (intended) flight path, self-organized or externally advised, cannot be maintained any longer. This may happen, for example, if a strong demanding factor suddenly step into action (e.g., engine failure). Consequently, the vehicle's actual flight path may quickly deviate from an expected norm, thus making a collision threat impending or unavoidable, say, in 5-15 seconds. During the FSTN-based prediction process, two subsets of flight paths are derived dynamically from the FSTN for each affected participant: dangerous (e.g., intersecting, or too close) and safe (non-intersecting and sufficiently separated). Then, this advisory information can be conveyed to the pilot or an automatic recovery system for guidance. An example of FSTN-based flight path prediction for two conflicting vehicles is schematically depicted in Fig. 2.

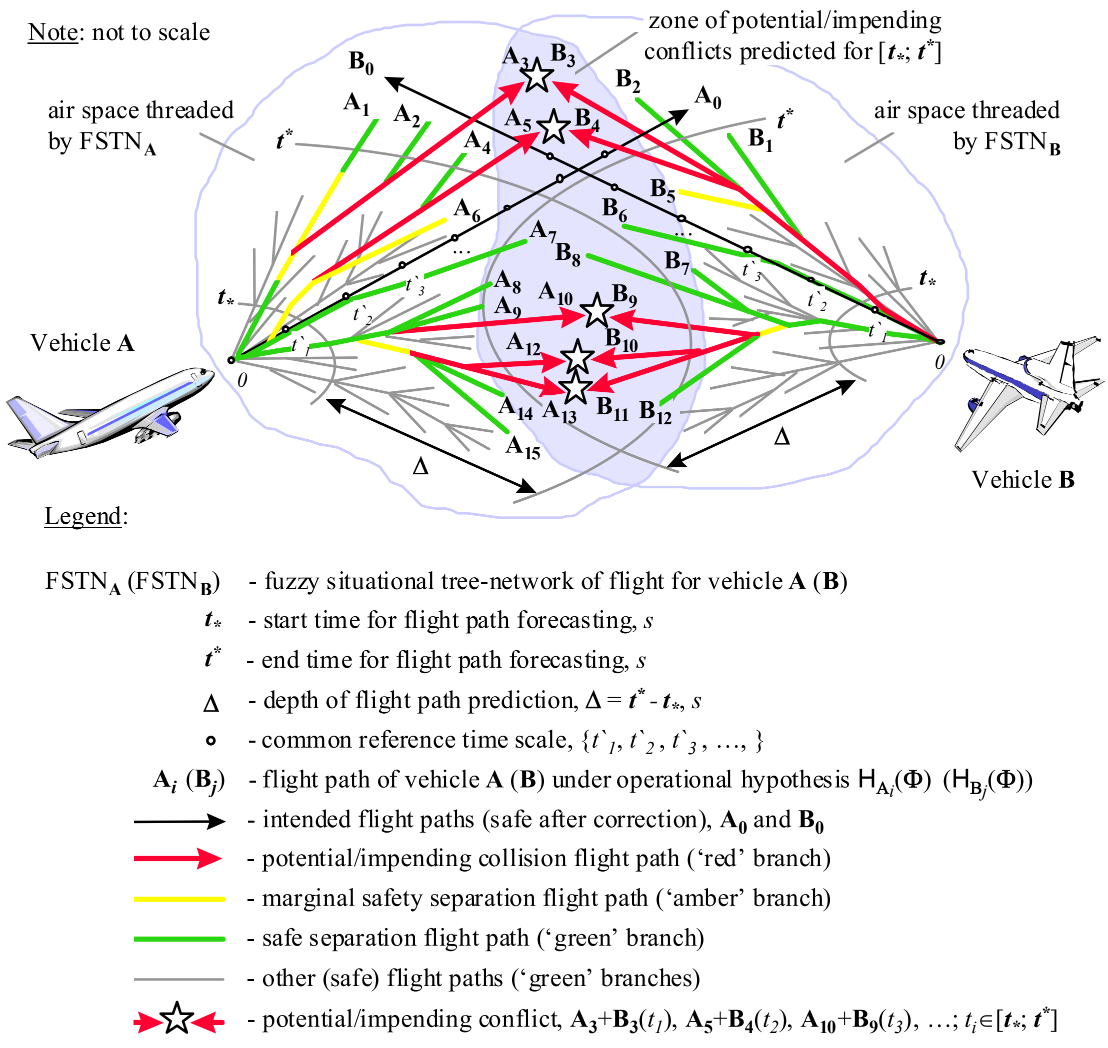


Figure 2: Flight path prediction for collision prevention/resolution in close air space

Pilot-vehicle intelligent interface—The number of alternative flight paths, both safe and unsafe, is beyond the capability of a human pilot to comprehend and process it in real time. Thus, the advisory information must be specially pre-processed. First, only a sub-tree of knowledge pertinent to the developing conflict and its 'neighborhood' is loaded from the FSTN for analysis. Second, this information must be displayed in intelligent, 'brain mapping' formats to be compatible with the pilot's internal model of flight. The underlying hypothesis is that pilots develop and store their situational (tactical) expertise in the form of tree [10]. However, due to human's constraints in information processing and memorization, pilots are often unable to build and use this knowledge base effectively under conditions of stress and uncertainty.

Therefore, the key task of pilot assistance in a FFS is to help the pilot access and analyze knowledge stored in the FSTN. The concepts of flight safety spectrum, Situational Forecast Display and other [8], [10], [16] can be used for this purpose. By means of intelligent formats, the pilot can compare various safe and unsafe paths as an integral picture.

5. Potential and Impending Conflicts

Two kinds of conflicts—As a result of combination of the methods introduced above, the developing AI model is expected to support two kinds of conflict management in close FF air space: (1) prevention of potential conflicts and (2) resolution of impending conflicts. These tasks are accomplished through self-organizing and knowledge-based processes, respectively.

Assumptions—There are three essential assumptions in the model:

1. Each vehicle constituting a FF group has permanent access to its global position data (e.g.: via GPS, GLONASS, etc.)
2. Each vehicle is equipped with a system for broadcasting 'personal data', including the current position, speed and flight direction (e.g., ADS-B [20]), as well as its intended flight path
3. Each vehicle is equipped with an identical FF support technology based on this model.

Potential conflict prevention—Prevention of potential conflicts is planned to accomplish through a self-organizing process of middle term flight planning (15-30 seconds ahead). It is based on the principles of collective behavior introduced above, a 3-D geometric model of a multi-vehicle FF environment [17]-[19], and real-time exchange of position and intended path data between the participants [20], [21]. However, a potential conflict may suddenly become an impending conflict if some unforeseen operational factors step into action and thus disturb the initially safe flight path.

Impending conflict resolution—Chances of collision will likely to remain even in a self-organized FF environment. Reynolds' simulation experiments demonstrate [13] that the collision rate in modeled over-crowded 'free-flight' situations in a bird flock ranges from 1 to 5 per 800 and 1000 simulation steps, respectively. To address this problem, physical what-if knowledge of the group dynamics is required. Near term predictions of possible flight paths, both unsafe and safe, can be carried out dynamically (5-15 seconds ahead) based on the information stored in the FSTN for each vehicle. One goal of this process is to advise the pilot of prohibited control inputs (controllable factors) and possible negative consequences of uncontrollable factors in a particular situation. Another goal is to recommend or employ automatically an evasive maneuver taking into account flight dynamics at the levels of a single vehicle and the whole group.

In the next section, main conceptual objects of the AI knowledge model are introduced. The emphasis is made on the second task (resolution of impending conflicts)

6. Main Objects

Introduction—Following is a system of concepts, or objects, used in the knowledge model. For each object there also exists a set of processing algorithms which are not described. However, some important algorithmic issues of the model will be discussed in the subsequent two sections.

Purpose—These concepts may serve as a formalized basis for designing algorithms and data structures of advanced AI technologies for conflict management in close FF air space.

Definition—An AI knowledge model of a complex FFS domain is a system, which includes the following objects:

$\{ t$; $\Omega(t)$; δ; $[t_*; t^*]$; $n(t)$; Δ; $\mathbf{V}$; $(\mathbf{A}; \mathbf{B}; \mathbf{C};\ldots)$; $\Omega(\mathbf{V})$; $n(\mathbf{V})$; N; V; $d_{\min}$; $d_{\mathbf{A},\mathbf{B}}$; $d_{i,j}$; V_D; V_S; $\boldsymbol{Cr}$; x; x_i; $\boldsymbol{x}$; $n(\boldsymbol{x})$; $\Omega(\boldsymbol{x})$; $\mathrm{R}(x_i)$; $\boldsymbol{x}(t)$; p; $\boldsymbol{x}_{\mathbf{A}}(t)$; $\underline{X}$; $\underline{x}(t)$; σ_k; $n(\underline{X})$; x_{inf}; x_{sup}; $\underline{x}$; $\underline{x}_i$; $\underline{\boldsymbol{x}}(t)$; $(\mathbf{V})$; $\boldsymbol{K}_{\mathbf{X}}$; $\boldsymbol{K}_{\mathbf{Y}}$; $\boldsymbol{K}_{\mathbf{Z}}$; $\boldsymbol{K}$; $\boldsymbol{D}_{\mathbf{X}}$; $\boldsymbol{D}_{\mathbf{Y}}$; $\boldsymbol{D}_{\mathbf{Z}}$; $\{ X; Y; Z \}$; $\mathbf{A}_0$; $(\mathbf{A})(\mathbf{B})$; $X_{(\mathbf{A})(\mathbf{B})}$; $Y_{(\mathbf{A})(\mathbf{B})}$; $Z_{(\mathbf{A})(\mathbf{B})}$; $d_{(\mathbf{A})(\mathbf{B})}$; $(X_{\mathbf{A}}; Y_{\mathbf{A}}; Z_{\mathbf{A}})$; $(X_{\mathbf{B}}; Y_{\mathbf{B}}; Z_{\mathbf{B}})$; $(\mathbf{A})+(\mathbf{B})$; $(\mathbf{V}_i)+(\mathbf{V}_j)$; $\Omega(\Phi)$; $\Omega_i(\Phi)$; $\Phi_{i\,j}$; $n(\Phi_{i\,j})$; $Nm(\Phi_{i\,j})$; $\Omega(\mathrm{L}(\Phi_{i\,j}))$; $x(\Phi_{i\,j})$, $\mathrm{L}(\Phi_{i\,j})$; {-VL, -L, -M, -S, -VS, O, +VS, +S, +M, +L, +VL}; Φ; Φ_{CON}; Φ_{UNC}; $\mathbf{E}$; $\mathbf{E}_i$; $\Omega(\mathbf{E})$; $n(\mathbf{E})$; $\Omega^{NR}(\mathbf{E})$; $\Omega^{JR}(\mathbf{E})$; $\Omega^{F}(\mathbf{E})$; $\Omega^{P}(\mathbf{E})$; Π; Π_j,; $\Omega(\Pi)$; $\{\mathbf{T}, \mathbf{O}, \mathbf{P}, \mathbf{W}, \mathbf{R}, \mathbf{Y}, \mathbf{B}, \mathbf{F},\ldots\}$; $n(\Pi)$; $\Omega^{NO}(\Pi)$; $\Omega^{O}(\Pi)$; $\Omega^{F}(\Pi)$; $\Omega^{CL}(\Pi)$; $\mathbf{s}$\`; $\mathbf{E}_*$, $\mathbf{E}^*$; $(\mathbf{E}_{i*}, \Pi_j, \mathbf{E}_k^*)$; $\mathbf{S}$; $\boldsymbol{M}$; $\mathrm{FSTN}_{\mathbf{A}}$; $\mathrm{FSTN}_{\mathbf{B}}$; $\boldsymbol{F}$; V_P; A_P; T_P; $\underline{\mathbf{A}}_i$; $\mathbf{A}_i$; $n(\mathrm{FSTN}_{\mathbf{A}})$; $\mathrm{H}_{\mathbf{A}i}(\Phi)$; $\Omega(\mathrm{H}|\mathbf{A})$; $\mathrm{H}_{\mathbf{A}0}(\Phi)$; $\mathbf{C}$; $n(\mathbf{C})$; $\{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$; $\mu_{\mathbf{C}}(\sigma_k)$; $(\mathbf{A})$; $\mathbf{A}+\mathbf{B}$; $\mathrm{H}_{\mathbf{A}i}(\Phi)+\mathrm{H}_{\mathbf{B}j}(\Phi)$; $\mathbf{A}+\mathbf{B}!$;

$\Omega(\mathbf{A}+\mathbf{B!})$; $\mathbf{A}_{\Uparrow}$; $t_{\Uparrow}$; $t|\mathbf{A}+\mathbf{B!}$; $\mathsf{T}_{\mathbf{A}\Uparrow}(\Phi)$; $\mathsf{T}_{\mathbf{A}\Uparrow}(\Phi)+\mathsf{T}_{\mathbf{B}\Uparrow}(\Phi)$; $\mathrm{A}_{\Downarrow}$; $Z|\mathbf{A}$; $Z^{-}|\mathbf{A}+\mathbf{B!}$; $Z^{+}|\mathbf{A}+\mathbf{B}$; … }

These objects are defined and explained below in the order as they appear in the set.

Time—The time parameter, t, does not require definition.

Common time scale—The common time scale, $\Omega(t)$, is an ordered set of evenly distributed time instants t_i, which are the same for all the vehicles involved in a FFS; $\Omega(t) = \{ t;\ t + \delta;\ t + 2\delta; ...; t_*; t_* + \delta;\ t_* + 2\delta;\ \ldots; t_* + (n(t)\text{-}1)\delta \}$; δ is the basic time increment, $\delta \in [0.5; 5]\ s$; and $n(t)$ is the total number of time instants in $\Omega(t)$. The common time scale is essential for synchronized processing and interpretation of information by all affected vehicles. The set $\Omega(t)$ for a particular vehicle is verified when the vehicle enters a FFS.

Flight prediction range—The flight prediction range, $[t_*; t^*]$, is a time range within which near-term predictions of vehicle's future flight paths are made; t_* and t^* are, respectively, the lower and upper time limits of the range. $[t_*; t^*]$ defines the depth of flight prediction via the FSTN.

Depth of flight path prediction—The depth of flight path prediction, Δ, is calculated as follows: $\Delta = t^* - t_*$. Practically, $\Delta \cong 5\div15$ s. Note also that $t_*, t^* \in \Omega(t)$.

FFS participant—The FFS participant (vehicle, participant), **V**, is a fixed- or rotary-wing aircraft or an airship, which performs navigation under FF or other rules [1] and which is present in the given close air space. Vehicles are also denoted by capital Latin letters in bold: **A**, **B**, **C**, …

Group of vehicles—N vehicles $\mathbf{V}_i$ participating in a FFS, $\Omega(\mathbf{V})$, form a group (system) of FFS participants, $\Omega(\mathbf{V}) = \{ \mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_{n(\mathbf{V})} \}$, or $\Omega(\mathbf{V}) = \{ \mathbf{A}, \mathbf{B}, \mathbf{C}, \ldots \}$, where $n(\mathbf{V}) \equiv N$. Note that $N = f(t)$.

Distance between two vehicles—The distance between two vehicles **A** and **B** ($\mathbf{V}_i$ and $\mathbf{V}_j$) in a group $\Omega(\mathbf{V})$, $d_{\mathbf{A},\mathbf{B}}$ or $d_{i,j}$, is the current distance between the centers of gravity of vehicles **A** and **B** ($\mathbf{V}_i$ and $\mathbf{V}_j$).

Close air space—In the model, the close (local) air space, V, is part of a three-dimensional Euclidean space, which accommodates the group $\Omega(\mathbf{V})$. V may have arbitrary shape. It also has a characteristic dimension, or distance, $d_{\min}$, where $d_{\min} \in [4; 6]$ km. A close air space may be dynamic or static.

Dynamic and static air space—The dynamic air space, V_D, is a close air space, which is created through and exists only during its association with a group of FFS participants. Its configuration and size are a function of time. It may be defined using the following criterion: $(\forall\mathbf{A}, \mathbf{B})$ $(\mathbf{A} \neq \mathbf{B}$ and $\mathbf{A}, \mathbf{B} \in \Omega(\mathbf{V}))$ $(\forall t)$ $(t \in \Omega(t))$ $((d_{\mathbf{A},\mathbf{B}} \leq d_{\min}$ or $\boldsymbol{Cr})$ and $(d_{\mathbf{A},\mathbf{B}} = f(t))$, where $\boldsymbol{Cr}$ is an alternative criterion (if any) for defining the meaning of a 'dynamic air space' and $d_{\min}$ is a constant.

Unlike V_D, the static air space, V_S, is linked to a particular geographic area (e.g., to an airport). In the latter case, $d_{\min}$ may be interpreted as a characteristic dimension of an ideal geometric model of V_S (e.g.: cylinder, sphere, ellipsoid, etc.).

Numeric flight variable—The numeric flight variable, x or x_i, is a measurable parameter, which describes a certain aspect of the 'pilot – vehicle – operational environment" system behavior as a real function of t. Numeric flight variables may be grouped into three classes: the pilot, the vehicle, and the operational environment.

Flight state vector—The flight state vector, $\boldsymbol{x}$, may be defined as an ordered list of all the numeric flight variables used to describe the behavior of the 'pilot – vehicle – operational environment" system: $\boldsymbol{x} = (x_1, x_2, \ldots, x_{n(x)})$, where $n(\boldsymbol{x})$ is the number of components in $\boldsymbol{x}$.

Vocabulary of flight variables—The vocabulary of flight variables, $\Omega(\boldsymbol{x})$, is an ordered list of names and other attributes of these variables. I.e.: $\Omega(\boldsymbol{x}) = \{ \mathrm{R}(x_1), \mathrm{R}(x_2), \ldots, \mathrm{R}(x_{n(x)}) \}$, where $n(\boldsymbol{x})$ is the number of records in $\Omega(\boldsymbol{x})$, or components in $\boldsymbol{x}$. The record $\mathrm{R}(x_i) = \{ Nm(x_i), Un(x_i), Fr(x_i), Min(x_i), Max(x_i), \ldots \}$ contains a set of key attributes of the variable, respectively, its name, physical unit, reference frame, minimal and maximal values and other.

Vehicle current state—The vehicle current state at a current time instant t, $t \in \Omega(t)$, is a vector $\boldsymbol{x}(t)$ of values of numeric flight variables measured at t: $\boldsymbol{x}(t) = (\boldsymbol{x}^1(t), \ldots, \boldsymbol{x}^p(t))$, where $p \equiv n(\boldsymbol{x})$. A current state of the vehicle **A** is denoted $\boldsymbol{x}_{\mathbf{A}}(t)$. Note that

$\boldsymbol{x}(t)$ includes the vehicle's current position, speed, flight direction, as well as the vehicle attitude and control surface positions, and other data. All participants of a FFS must broadcast data on their current state $\boldsymbol{x}(t)$.

Fuzzy measurement scale—The fuzzy measurement scale (Fig. 3), $\underline{X}$, is a finite set of fuzzy set-values, $\underline{X} = \{\sigma_1, \ldots, \sigma_{n(\underline{X})}\}$, which are used to approximate a numeric value $x(t)$ of a numeric flight variable x (or x_i) by its 'fuzzy equivalent' – a fuzzy set $\underline{x}(t)$, where $\underline{x}(t) \cong \sigma_k$, $\sigma_k \in \underline{X}$. Note that in the flight research domain, the number of fuzzy values in $\underline{X}$, $n(\underline{X})$, is between 5 and 15. The following generic criterion for fuzzy value recognition can be used to implement the fuzzy measurement mapping $x \rightarrow \underline{x}$: $(\forall x(t))\ (x(t) \in [x_{inf}; x_{sup}])\ (\exists \sigma_k)\ (\sigma_k \in \underline{X})\ (\mu_{\sigma k}(x(t)) = max\ \mu_{\sigma_i}(x(t));\ i = 1, ..., n(\underline{X}),\ k \in \{1, ..., n(\underline{X})\}) \Rightarrow (\underline{x}(t) \cong \sigma_k)$. More discussion on the role of fuzzy measurements in flight analysis can be found in [10].

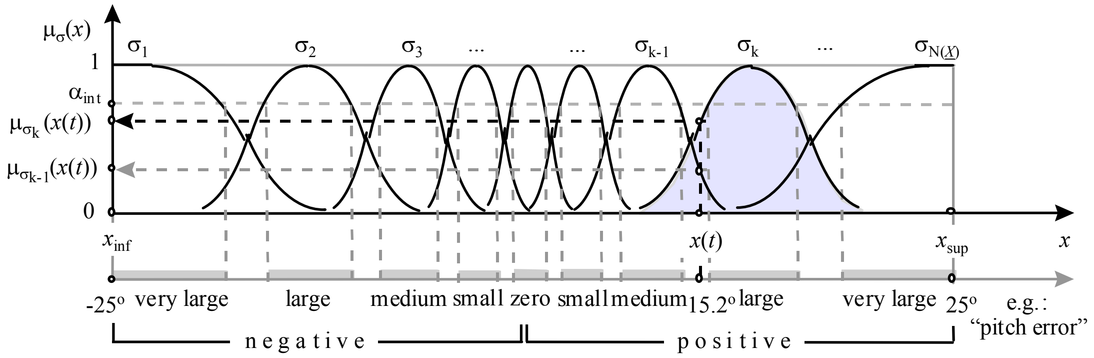


Figure 3: Fuzzy measurement scale of a linguistic flight variable

Linguistic (flight) variable—The linguistic flight variable, $\underline{x}$ or $\underline{x}_i$, is a variable, which measures flight states by means of fuzzy set-values from a fuzzy measurement scale $\underline{X}$ based on numeric values of a corresponding variable x. In other words, the variable $\underline{x}$ is assigned that fuzzy value $\underline{x}(t)$ from $\underline{X}$, $\underline{x}(t) \cong \sigma_k$, which best represents $x(t)$ using the recognition criterion introduced above. Through this process, the state of the 'pilot – vehicle – operational environment" system can be assessed approximately but in mathematically correct terms. By fuzzifying the system state space it becomes possible to describe a sufficiently large situational domain of flight without experiencing the 'curse of dimensionality'. Note that $\underline{\boldsymbol{x}} = (\underline{\boldsymbol{x}}^1, \ldots, \underline{\boldsymbol{x}}^p)$ and $\underline{\boldsymbol{x}}(t) = (\underline{\boldsymbol{x}}^1(t), \ldots, \underline{\boldsymbol{x}}^p(t))$.

Protecting air space ('safety bubble')—The 'safety bubble', $(\mathbf{V})$, is an imaginary ellipsoid or sphere, which encapsulates the vehicle $\mathbf{V}$. Its center is located in the vehicle's center of gravity and three axes of the bubble coincide with the vehicle's body axes (Fig. 4). Note that other vehicle's 'safety bubble' or an obstacle cannot infringe $(\mathbf{V})$ under any operational conditions. This is basically the purpose of $(\mathbf{V})$ – to protect the vehicle's surrounding space from penetration with a sufficient safety margin. The relative dimensions of $(\mathbf{V})$ and $\mathbf{V}$ ('protection factors') along the vehicle's body axes X, Y, and Z, $\boldsymbol{K}_i$, are calculated as follows: $\boldsymbol{K}_i = [(\mathbf{D}_i/\mathbf{d}_i) - 1]/2$, where $\mathbf{D}_i$ and $\mathbf{d}_i$ are the characteristic dimensions of $(\mathbf{V})$ and $\mathbf{V}$ along the axis $\boldsymbol{i}$, $\boldsymbol{i} \in \{ X; Y; Z \}$. Note, when $\boldsymbol{K} = \boldsymbol{K}_X = \boldsymbol{K}_Y = \boldsymbol{K}_Z$, the safety space margin around $\mathbf{V}$ is equal to $\mathbf{d}_i$ times $\boldsymbol{K}$, $\boldsymbol{i} \in \{ X; Y; Z \}$. For a close air space, probably $\boldsymbol{K}_{min} \in [5; 15]$, see Fig. 4.

Intended (nominal) flight path—The intended (nominal) flight path of the vehicle $\mathbf{A}$, $\mathbf{A}_0$, is a safe flight path derived from the process of self-organizing collective behavior (Fig. 2), or other sources. It is calculated and communicated to the rest of the group in the following format: $\mathbf{A}_0 = \{ \boldsymbol{x}(t_1), \ldots, \boldsymbol{x}(t_k), \ldots, \boldsymbol{x}(t_{n(t)}); \underline{\boldsymbol{x}}(t_1), \ldots, \underline{\boldsymbol{x}}(t_k), \ldots, \underline{\boldsymbol{x}}(t_{n(t)}) \}$, where $n(t)$ is the number of sample points in $\mathbf{A}_0$ and $t_k \in \Omega(t)$. The intended flight path may be corrected to avoid potential or impending conflicts; in this case it may also include bypass segments (the definition will follow).

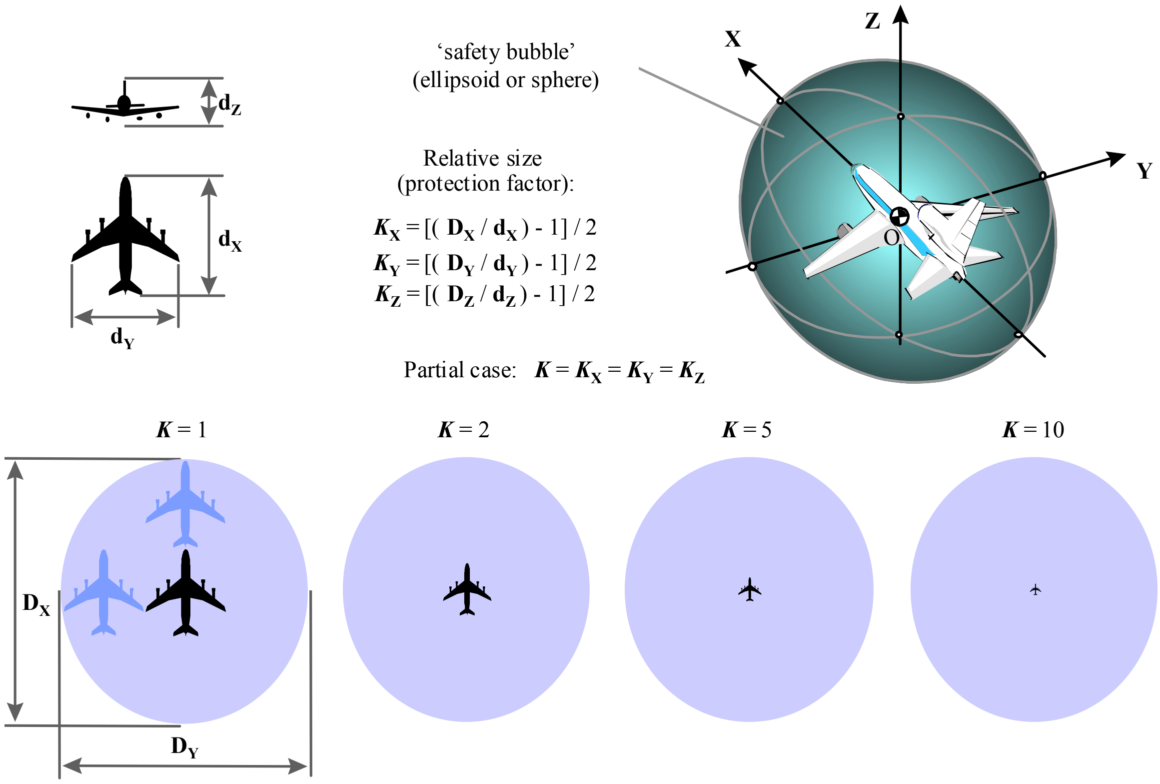


Figure 4: Flight vehicle and its protecting air space ('safety bubble')

Tangent position of 'safety bubbles'—The tangent position of 'safety bubbles' of the vehicles **A** and **B**, (**A**)(**B**), is a relative position of (**A**) and (**B**) in *V*, when (**A**) is tangent to (**B**) in a point with coordinates ($X_{(A)(B)}$; $Y_{(A)(B)}$; $Z_{(A)(B)}$). The distance between the two vehicles at (**A**)(**B**) is $d_{(A)(B)}$; it is a function of the vehicles' attitude, relative position in space – ref. relevant solid geometry formulas in [17]. The coordinates of the vehicles' centers of gravity are (X_A; Y_A; Z_A) and (X_B; Y_B; Z_B).

Penetration of 'safety bubbles'—The penetration of 'safety bubbles' for two vehicles **A** and **B** (or $\mathbf{V}_i$ and $\mathbf{V}_j$) is such a relative position of the two vehicles, denoted (**A**)+(**B**) or ($\mathbf{V}_i$)+($\mathbf{V}_j$), when the distance between **A** and **B** becomes smaller than $d_{(A)(B)}$, i.e. $d_{A,B} < d_{(A)(B)}$. In other words, at (**A**)+(**B**) the vehicles' 'safety bubbles' have a non-empty intersection: $(\mathbf{A})\cap(\mathbf{B}) \neq \varnothing$. This condition means that the vehicles flight safety is compromised.

Anticipated operational conditions—The anticipated operational conditions (factors) of flight is a set $\Omega(\Phi)$, $\Omega(\Phi) = \{ \Omega_1(\Phi), \ldots, \Omega_{n(\Phi)}(\Phi) \}$, where $\Omega_i(\Phi)$ is a set-specification of *i*-th class of homogeneous operational factors. The specification $\Omega_i(\Phi)$ has the following format: $\Omega_i(\Phi) = \{ [Nm(\Phi_{i\,1}), \Omega(L(\Phi_{i\,1}))], [Nm(\Phi_{i\,1}), \Omega(L(\Phi_{i\,1}))], \ldots, [Nm(\Phi_{i\,n(\Phi)}, \Omega(L(\Phi_{i\,n(\Phi)}))] \}$. $Nm(\Phi_{i\,j})$ is the name of *j*-th sub-class within *i*-th class; $\Omega(L(\Phi_{i\,j}))$ is a fuzzy measurement scale used to measure the level, $L(\Phi_{i\,j})$, of the factor $\Phi_{i\,j}$: $\Omega(L(\Phi_{i\,j})) = \{$ -VL, -L, -M, -S, -VS, 0, +VS, +S, +M, +L, +VL $\}$, where + ≡ 'positive', - ≡ 'negative', V ≡ 'very', L ≡ 'large', M ≡ 'medium', S ≡ 'small', and 0 ≡ 'about zero'.

Operational condition—The operational condition (factor), Φ, is a triplet of attributes: $\Phi = (Nm(\Phi_{i\,j}), Nm(\Phi_{i\,j}), L(\Phi_{i\,j}))$, where: $Nm(\Phi_{i\,j})$ is the name of a sub-class (i, j); $Nm(\Phi_{i\,j})$ is the name of a flight variable or a vector of flight variables, which represents Φ, $Nm(\Phi_{i\,j}) \in \Omega(\boldsymbol{x})$; $L(\Phi_{i\,j})$ is the fuzzy level of a base variable applied in the condition Φ.

Controllable and uncontrollable (operational) factor—The controllable operational factor, Φ_{CON}, is an operational factor, which negative effects can be managed in a FFS by the pilot. The uncontrollable factor, Φ_{UNC}, is a factor, which adverse effects cannot be rectified by the pilot or an automatic system. Examples of uncontrollable factors are critical engine failure and strong wind shear, etc.

Flight event—The flight event (event), **E**, is a characteristic state of the "pilot – vehicle – operational environment" system. Flight events are considered as special "points", or nodes, in a multi-dimensional situational space. They are important to the pilot in terms of flight logic planning or execution. Examples of flight events are as follows: $\mathbf{E}_{21}$: "engine #1 failed", $\mathbf{E}_{7}$: "low airspeed", $\mathbf{E}_{1}$: "bank angle within 25^{o}-30^{o}", $\mathbf{E}_{16}$: "go-around decision", $\mathbf{E}_{9}$: "altitude 1,000 ft", $\mathbf{E}_{15}$: "touchdown", $\mathbf{E}_{11}$: "heading 175^{o}". Events stand for discrete components of the flight situation model.

Calendar of flight events—A complete set of events, $\Omega(\mathbf{E})$, which may occur during a certain phase of flight, is called the flight event calendar, $\Omega(\mathbf{E}) = \{\ \mathbf{E}_1, \ldots, \mathbf{E}_{n(\mathbf{E})}\ \}$. It forms a logical framework of a pilot's situational decision making, automatic flight control, and a situation itself. During flight, all the events from $\Omega(\mathbf{E})$ can be grouped dynamically according to their current transition state into the following subsets: $\Omega(\mathbf{E}) = \Omega^{NR}(\mathbf{E}) \cup \Omega^{JR}(\mathbf{E}) \cup \Omega^{F}(\mathbf{E}) \cup \Omega^{P}(\mathbf{E})$. These subsets represent, respectively, 'not recognized', 'just recognized', 'frozen' and 'past' events [6].

Flight process—Unlike the flight event, the flight process, Π, is a continuous component of the flight situation model. It represents a distinctive lasting aspect (action, factor, function, input, etc.) of the system's behavior. Depending on the physical nature, processes may be divided into three main groups (ref. [6], [7], [15] for more detail): pilot's tactical (situational) decision making and pilot errors - "piloting task" (**T**), system "state observer" (**O**), "control procedure" (**P**), and some other; external operational conditions – "wind" (**W**), "rain" (**R**), "runway surface condition" (**Y**); and onboard system functions and system malfunctions - "function" (**B**) and "failure" (**F**).

Flight process examples are as follows. $\mathbf{T}_8$: "perform a coordinated right turn at a 25^{c} bank", $\mathbf{O}_6$: "observe bank angle and roll rate", $\mathbf{P}_5$: "flaps–down, from 0^{o} to 30^{o}", $\mathbf{W}_1$: "strong wind shear, accident *mm/dd/yy*", $\mathbf{R}_2$: "tropical shower of a trapezoid profile with the maximum intensity of 300 mm/hr", $\mathbf{Y}_3$: "wet runway", $\mathbf{B}_1$: "yaw SCAS inoperative", $\mathbf{F}_{19}$: "rudder hardover to $+25^{o}$".

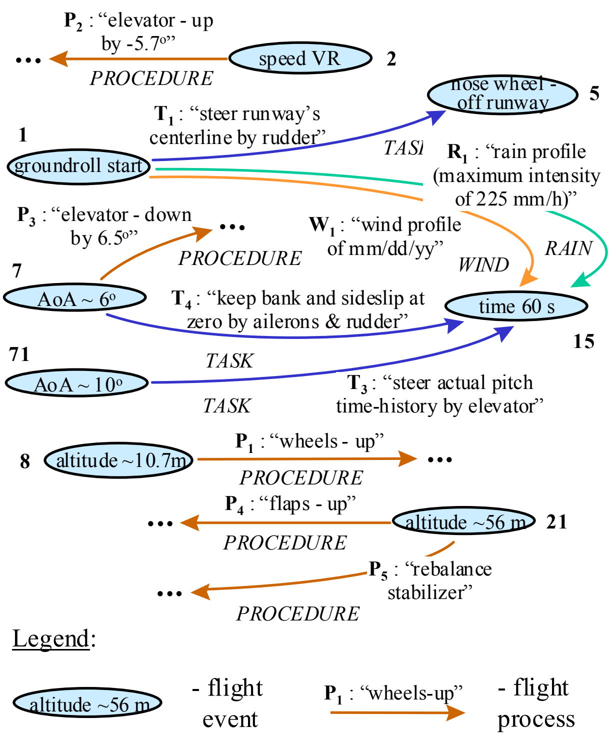


Figure 5: Flight accident scenario "Takeoff under severe microburst conditions"

United list of flight processes—All processes, planned for a flight situation or a group of situations, constitute a united list of flight processes $\Omega(\Pi)$, $\Omega(\Pi) = \{\ \Pi_1, \ldots, \Pi_{n(\Pi)}\ \}$, $\Pi \in \{\ \mathbf{T}, \mathbf{O}, \mathbf{P}, \mathbf{W}, \mathbf{R}, \mathbf{Y}, \mathbf{B}, \mathbf{F}, \ldots\ \}$. At any time during flight all flight processes from $\Omega(\Pi)$ can be grouped according to their current state into the following subsets: $\Omega(\Pi) = \Omega^{NO}(\Pi) \cup \Omega^{O}(\Pi) \cup \Omega^{F}(\Pi) \cup \Omega^{CL}(\Pi)$, which are, respectively 'not open', 'open', 'frozen', and 'closed' processes [6].

Elementary situation—Every process Π_j runs between two events, the "source" event and the "target" event. The source event, $\mathbf{E}_{i*}$, opens Π_j, whilst the target event, $\mathbf{E}_k^{*}$, closes it in flight, $\mathbf{E}_{i*}, \mathbf{E}_k^{*} \in \Omega(\mathbf{E})$. A triplet **s`**, **s`** = ($\mathbf{E}_{i*}$, Π_j, $\mathbf{E}_k^{*}$), is called the elementary situation - ref. ($\mathbf{E}_9$, $\mathbf{T}_8$, $\mathbf{E}_{11}$) in the examples above.

Flight (situation) scenario—The flight (situation) scenario, **S**, is a plan for implementing this situation and associated piloting tactics in simulation or operation. A flight scenario may be depicted as a directed graph: $\mathbf{S} = \Omega(\mathbf{E}) \cup \Omega(\Pi)$. Its events (vertices), $\Omega(\mathbf{E})$, and directed processes (arcs), $\Omega(\Pi)$, are linked together forming a logical cause-and-effect model of this situation. Note that a flight scenario may be viewed as a union of its elementary situations. Flight scenarios capture cause-and-effect and other key relationships between discrete and continuous elements of flight, thus mapping its invariant structure. Fig. 5 depicts a realistic scenario **S** of an accident with a transport airplane, titled "Takeoff under severe microburst conditions" [6], [10].

Situational model of flight—This is a system of algorithms and data structures, ***M***, which model the behavior of the 'pilot – vehicle – operational environment' system in a complex flight situation. A formal relationship for executing a flight scenario in simulation can be defined as follows: $(\forall \mathbf{S})\ (\mathbf{S} = \Omega(\mathbf{E}) \cup \Omega(\Pi))\ (\exists \mathbf{s})\ (\mathbf{s}` = (\mathbf{E}_i, \mathbf{E}_k, \Pi_j)\ (((\mathbf{E}_i \in \Omega^P(\mathbf{E}) \wedge \mathbf{E}_k \notin \Omega^P(\mathbf{E}) \wedge \Pi_j \notin \Omega^{CL}(\Pi)) \wedge (t \geq t[\mathbf{E}_i \in \Omega^P(\mathbf{E})]+\tau)) \Rightarrow \Pi_j \in \Omega^A(\Pi)) \vee ((\mathbf{E}_k \in \Omega^P(\mathbf{E}) \Rightarrow \Pi_j \in \Omega^{CL}(\Pi))$, where τ is a delay in event recognition. This relationship, together with generic models of flight events and flight processes, constitute a generic computational algorithm of ***M***.

Fuzzy situational tree-network of flight—The concept of fuzzy situational tree-network, or FSTN, is schematically shown in Fig. 6 (ref. [6], [8]-[10] for more detail).

A series of flights exploring a close air space under non-standard conditions can be simulated using the autonomous situational model of flight and a library of flight situation scenarios. The resulting flight paths for the vehicle **A** can be represented as a tree, $\text{FSTN}_\mathbf{A}$, which fuzzy branches deviate from the intended path $\mathbf{A_0}$ *if* a certain operational factor or a combination of factors step into action. For this reason such virtual flight trajectories are called 'what-if' paths. A crown of the $\text{FSTN}_\mathbf{A}$ can be specially shaped to thread a FFS domain under examination, from initially safe states towards flight

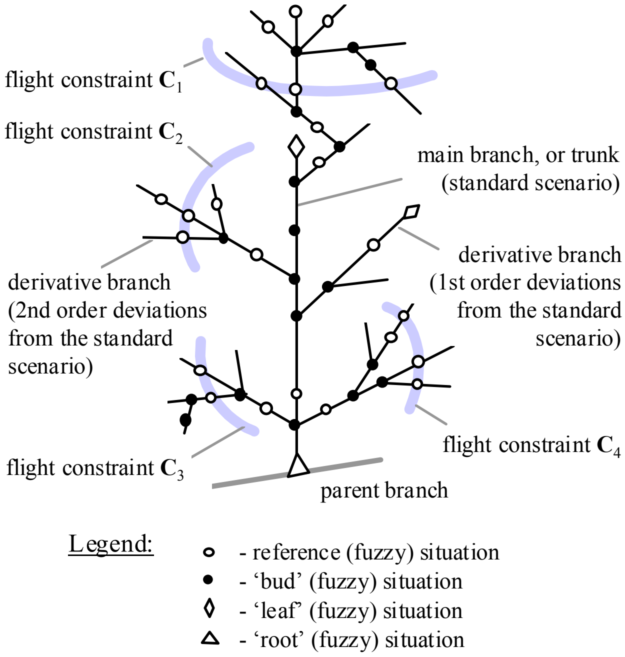


Figure 6: Fuzzy situational tree-network of flight

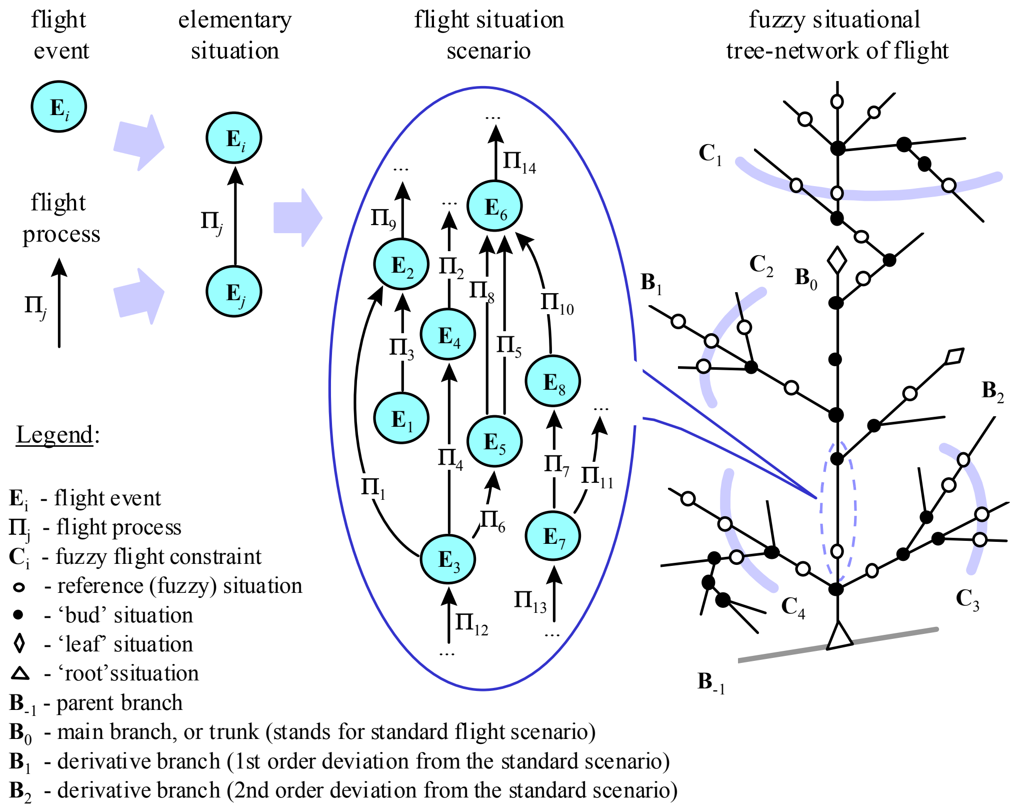


Figure 7: Relationship between the micro- and macro-structures of flight

constraints. Each fuzzy branch represents a path, which incorporates the effect of a special operational hypothesis (see below). The goal of branching is to reveal zones of possible conflicts under various demanding conditions based on the physics of flight.

Micro- and macro-structures of flight—Thus, knowledge of a complex FFS domain can be formalized on two interrelated levels [6]: the micro-level and the macro-level. The micro-level is represented by the autonomous flight situation model, which is based on the notions of flight event, flight process, elementary situation, and flight scenario. The macro-level is represented in the form of fuzzy situational tree-network of flight. The relationship between these two levels is graphically depicted in Fig. 7.

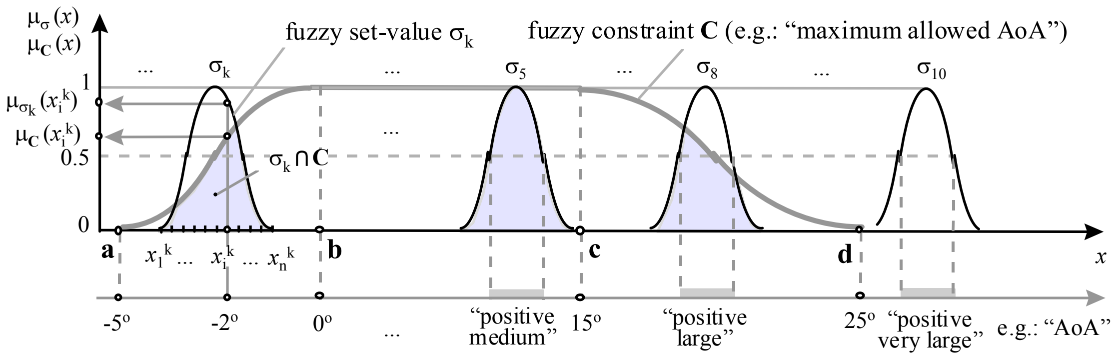
Intelligent format—The topology, structure and safety characteristics of the FSTN can be used to develop knowledge mapping formats for pilot-vehicle intelligent interface. The intelligent format, $\boldsymbol{F}$, is a mapping $\boldsymbol{F}$: ($\text{FSTN}_{\mathbf{A}}$, $\boldsymbol{x}(t)$, $\underline{\boldsymbol{x}}(t)$, …) → $V_P \times A_P \times T_P \times \ldots$, where V_P, A_P, T_P, … are, respectively, visual, audio-, tactile and other sensory spaces of the human pilot, and × stands for Cartesian product. The purpose of these formats is to intelligently convey knowledge of a complex flight situation sub-domain to the pilot, e.g. in an emergency. It is believed that these representations will be more compatible with the pilot's internal (mental) model of flight.

Fuzzy flight path-branch—The fuzzy flight path-branch, $\underline{\mathbf{A}}_i$ or $\mathbf{A}_i$, is a cause-and-effect chain of fuzzy states in the $\text{FSTN}_{\mathbf{A}}$ linked by fuzzy transitions (see Fig. 2 and Fig. 6): $\underline{\mathbf{A}}_i = \{ \underline{\boldsymbol{x}}(t_\Delta), \ldots, \underline{\boldsymbol{x}}(t_i), \ldots, \underline{\boldsymbol{x}}(t_\Diamond); \ldots \}$. Therefore, $\text{FSTN}_{\mathbf{A}}$ can be considered as a collection of interconnected branches $\mathbf{A}_i$, $\text{FSTN}_{\mathbf{A}} = \{ \mathbf{A}_0, \ldots, \mathbf{A}_i, \ldots, \mathbf{A}_{n(\text{FSTN}\mathbf{A})\text{-}1} \}$, where $n(\text{FSTN}_{\mathbf{A}})$ is a total number of branches in $\text{FSTN}_{\mathbf{A}}$. Note that time instants t_i in $\mathbf{A}_i$ may not belong to $\Omega(t)$.

Operational hypothesis—The operational hypothesis for a branch-path $\mathbf{A}_i$, $\mathsf{H}_{\mathbf{A}i}(\boldsymbol{\Phi})$, is modeled as a subset of demanding operational conditions, which are examined along $\mathbf{A}_i$. In other words, $\mathsf{H}_{\mathbf{A}i}(\boldsymbol{\Phi})$ is an operational hypothesis used to 'implant' $\mathbf{A}_i$ into the $\text{FSTN}_{\mathbf{A}}$.

Vehicle's operational space—A list of all the operational hypotheses for the vehicle $\mathbf{A}$ is denoted as $\Omega(\mathsf{H}|\mathbf{A})$, $\Omega(\mathsf{H}|\mathbf{A}) = \{ \mathsf{H}_{\mathbf{A}0}(\boldsymbol{\Phi}), \mathsf{H}_{\mathbf{A}1}(\boldsymbol{\Phi}), \ldots, \mathsf{H}_{\mathbf{A}i}(\boldsymbol{\Phi}), \ldots, \mathsf{H}_{n(\mathbf{A}i)}(\boldsymbol{\Phi}) \}$, where $\mathsf{H}_{\mathbf{A}0}(\boldsymbol{\Phi})$ corresponds to a intended (safe) flight path $\mathbf{A}_0$. Note that there may be more than one branch in the $\text{FSTN}_{\mathbf{A}}$, that has been constructed under the same hypothesis, i.e. $(\exists i, j)\ (i \neq j)\ (i, j \in [0; n(\text{FSTN}_{\mathbf{A}})])\ (\mathsf{H}_{\mathbf{A}i}(\boldsymbol{\Phi}) = \mathsf{H}_{\mathbf{A}j}(\boldsymbol{\Phi}))$.

Fuzzy flight constraint—The fuzzy flight constraint of a variable x is a fuzzy set $\mathbf{C}$ built over the universe of discourse of the numeric variable x. $\mathbf{C}$ is defined by four reference points $\{ \mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d} \}$ of its numeric carrier (Fig. 8). The purpose of fuzzification of a system of the vehicle's operational constraints $\Omega(\mathbf{C})$, $\Omega(\mathbf{C}) = \{ \mathbf{C}_1, \ldots, \mathbf{C}_i, \ldots, \mathbf{C}_{n(\mathbf{C})} \}$, is to account for the uncertainties of our knowledge of the vehicle's actual flight envelope.



Degree of compatibility of fuzzy contraint $\mathbf{C}$ and fuzzy value σ_k:

$$\mu_{\mathbf{C}}(\sigma_k) = \frac{\text{card}(\sigma_k \cap \mathbf{C})}{\text{card}(\sigma_k)}, \text{ where } \text{card}(\sigma_k \cap \mathbf{C}) = \sum_{i=1}^{n} \boldsymbol{min}\,[\mu_{\sigma_k}(x_i^k), \mu_{\mathbf{C}}(x_i^k)],\ \text{card}(\sigma_k) = \sum_{i=1}^{n} \mu_{\sigma_k}(x_i^k)$$

Figure 8: Fuzzy constraint and fuzzy values of a linguistic flight variable

Within the FSTN structure, a fuzzy constraint may be considered as an external object, or a strip, attached to its one or several branches (see Fig. 6). The position of a fuzzy constraint within the FSTN can be revealed only during FSTN construction. The degree of compatibility of a fuzzy state and fuzzy constraint, $\mu_C(\sigma_k)$, is measured using the operation of intersection for fuzzy sets. This measure may be used to assess the degree of danger of alternative fuzzy flight paths.

Safety corridor—The safety corridor for the vehicle **A**, (**A**), is a surface in V formed by the vehicle's 'safety bubble' when it follows the intended flight path (Fig. 9).

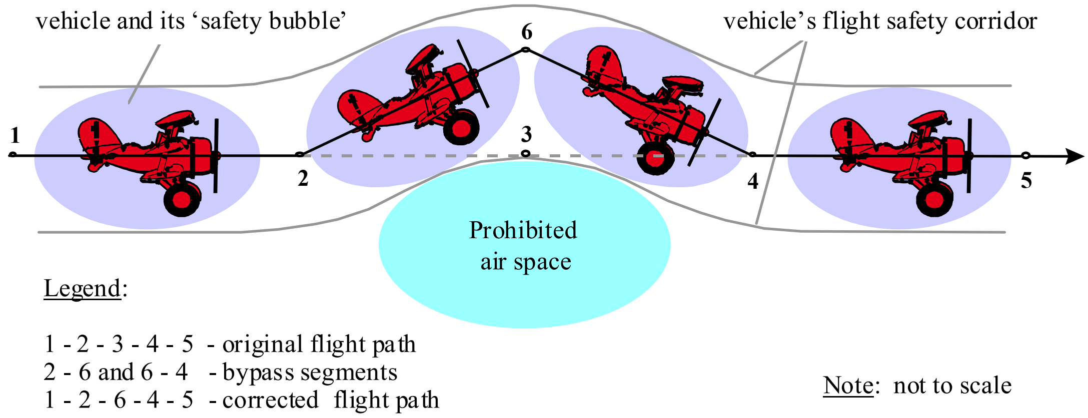


Figure 9: Prohibited air space and a bypass trajectory

Bypass segment—The bypass segment is a short trajectory of a special geometric profile used to bypass a protecting or prohibited air space or for recovery. Examples of bypass segments are depicted in Fig. 10.

Potential conflict—The potential conflict between two vehicles **A** and **B**, **A**+**B**, means that their safety bubbles along the intended fight paths may intersect in the near future, if and only if a certain combination (pair) of operational hypotheses, $\mathsf{H}_{\mathbf{A}i}(\Phi)+\mathsf{H}_{\mathbf{B}j}(\Phi)$, becomes true (i.e. takes place in the flight). In other words, there exist two virtually intersecting fuzzy flight paths in FSTN$_{\mathbf{A}}$ and FSTN$_{\mathbf{B}}$, which have the same fuzzy position in V and the same or close occurrence time. A notional example of potential conflict resolution is shown in Fig. 11.

Impending conflict—The impending (about-to-happen) conflict between two vehicles **A** and **B**, **A**+**B!**, is a potential conflict, which is very likely to occur in a few seconds. In other words, a conflict is impending if some operational hypothesis pair from $\Omega(\mathsf{H}|\mathbf{A})\cup\Omega(\mathsf{H}|\mathbf{B})$ becomes true, i.e. actually present in the flight. A current subset of impending conflicts is denoted by $\Omega(\mathbf{A}+\mathbf{B!})$, where $\Omega(\mathbf{A}+\mathbf{B!}) = f(t)$. Monitoring and prediction of impending conflicts are carried out by means of the vehicles' fuzzy situational tree-networks.

Note: an impending conflict can be avoided if there exists (and then - followed) at least one child branch[2], $\mathbf{A}_{\Uparrow}$, in FSTN$_{\mathbf{A}}$ or/and one child branch, $\mathbf{B}_{\Uparrow}$, in FSTN$_{\mathbf{B}}$ implanted at $t_{\Uparrow}$, $t_{\Uparrow} > t$ and $t_{\Uparrow} \in [t_*; t^*]$, so that the vehicles' 'safety bubbles' tracking the path $\mathbf{A}_{\Uparrow}$ or/and $\mathbf{B}_{\Uparrow}$ do not intersect in V, i.e. $d_{\mathbf{A},\mathbf{B}} > d_{(\mathbf{A})(\mathbf{B})}$, at about the

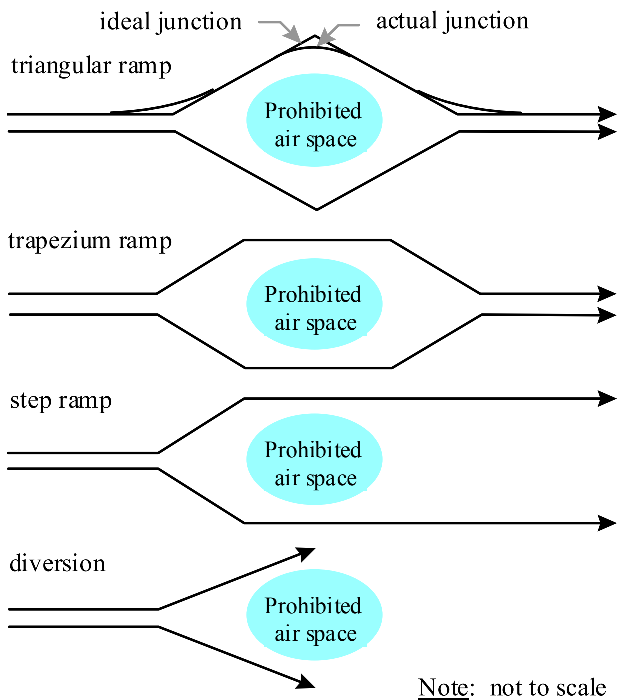


Figure 10: Examples of bypass trajectories

[2] the 'child branch' is referred to a current fuzzy branch, which leads to the impending conflict

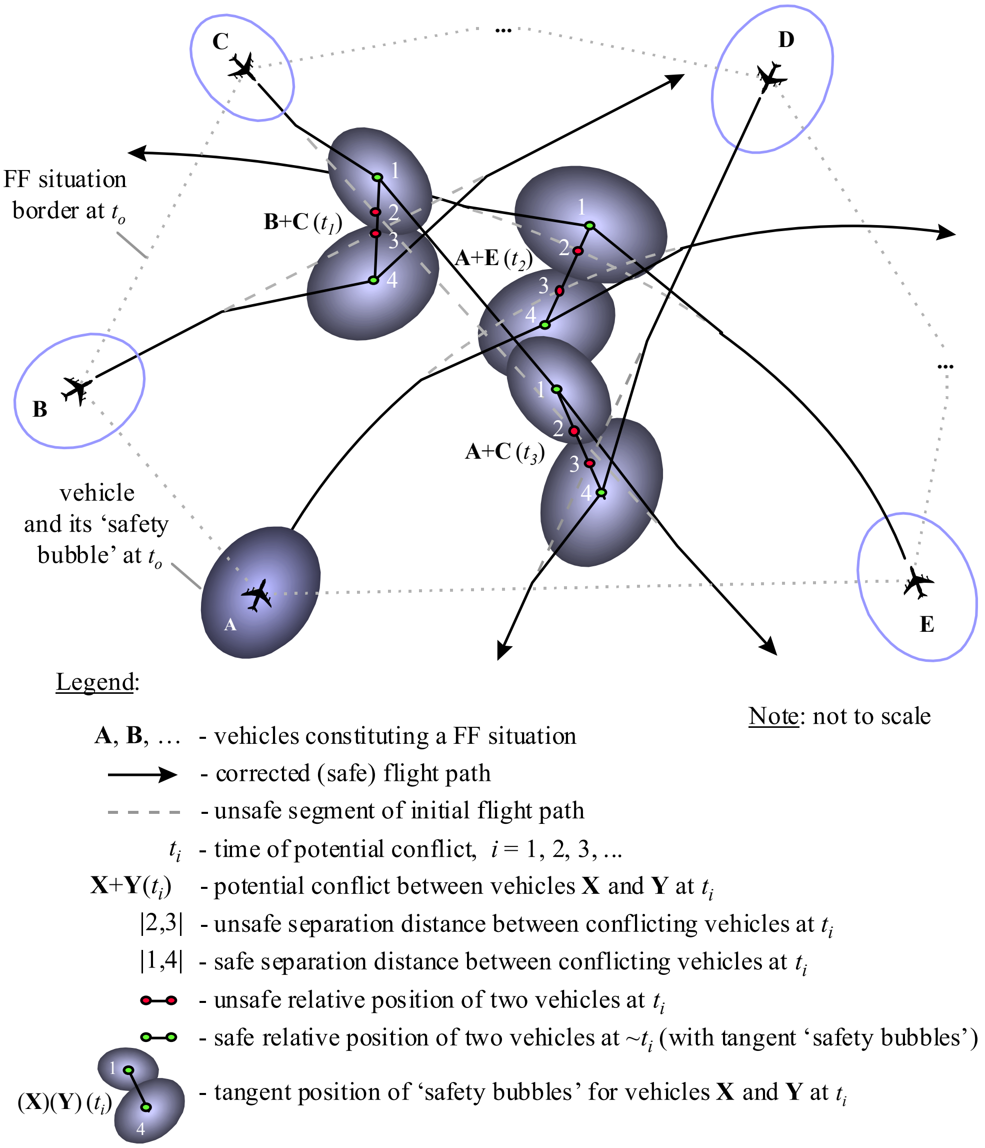


Figure 11: Potential conflict resolution (example)

expected collision time $t|\mathbf{A}+\mathbf{B}!$. (see the definition of $\mathbf{A}_{\Uparrow}$ below).

Joint recovery tactics—A pair of operational hypotheses $\mathsf{H}_{\mathbf{A}i}(\Phi)+\mathsf{H}_{\mathbf{B}j}(\Phi)$, which correspond to the recovery paths for the conflicting vehicles **A** and **B**, $\mathbf{A}_{\Uparrow}$ and $\mathbf{B}_{\Uparrow}$, is called the joint recovery tactics, or joint recovery maneuver, $\mathsf{T}_{\mathbf{A}\Uparrow}(\Phi)+\mathsf{T}_{\mathbf{B}\Uparrow}(\Phi)$. Note: normally, the implementation of a single tactics or maneuver for the vehicle **A** or **B**, i.e. $\mathsf{T}_{\mathbf{A}\Uparrow}(\Phi)$ or $\mathsf{T}_{\mathbf{B}\Uparrow}(\Phi)$, should suffice to resolve an impending conflict between **A** and **B**. However, in the model, adjustment of the flight paths for both vehicles are considered as more appropriate for the sake of the overall safety and resources economy in the group $\Omega(\mathbf{V})$.

Recovery flight path—The recovery flight path for the vehicle **A**, $\mathrm{A}_{\Uparrow}$, is a safe fuzzy flight path or a concatenation of segments of safe fuzzy flight paths from $\mathrm{FSTN}_{\mathbf{A}}$, which corresponds to the recovery tactics $\mathsf{T}_{\mathbf{A}\Uparrow}(\Phi)$.

Irreversible flight path—The irreversible flight path for vehicle **A**, $\mathrm{A}_{\Downarrow}$, is a fuzzy flight path from the $\mathrm{FSTN}_{\mathbf{A}}$, which includes an impending collision situation and there is no alternative fuzzy path in $\mathrm{FSTN}_{\mathbf{A}}$ for avoiding the conflict.

Scanned air space—The scanned (monitored, checked) air space for the vehicle **A**, $Z|\mathbf{A}$, is a part of close air space V, which is threaded by fuzzy flight paths from $\mathrm{FSTN}_{\mathbf{A}}$ ahead of **A** for all t_i, $t_i \in [t_*;\ t^*]$. These path-branches are constructed by applying hypotheses from the vehicle's operational space $\Omega(\mathsf{H}|\mathbf{A})$. Thus, the size, shape and orientation of $Z|\mathbf{A}$ in V, as well its extent in time, depend on three parameters: (1) current time, t; (2) flight prediction range, $[t_*; t^*]$; and (3) set of operational hypotheses selected for monitoring, $\Omega(\mathsf{H}|\mathbf{A})$; i.e. $Z|\mathbf{A} = f(t; [t_*; t^*]; \Omega(\mathsf{H}|\mathbf{A}))$.

Prohibited zone—The prohibited zone for vehicles **A** and **B** (see Fig. 2 and Fig. 9), $Z^-|\mathbf{A}+\mathbf{B}!$, is a subset of fuzzy points in space V, which correspond to all impending conflicts between **A** and **B**. Note that $Z^-|\mathbf{A}+\mathbf{B}! \subset Z|\mathbf{A} \cap Z|\mathbf{B}$ and $\mathbf{A}+\mathbf{B}! \in Z^-|\mathbf{A}+\mathbf{B}!$.

Allowed maneuver zone—Unlike $Z^-|\mathbf{A}+\mathbf{B}!$, the allowed maneuver zone, $Z^+|\mathbf{A}+\mathbf{B}$, is a zone in V, which does not contain impending conflicts, i.e. $Z^+|\mathbf{A}+\mathbf{B} \subset Z|\mathbf{A} \cap Z|\mathbf{B}$ (see Fig. 2), but $\mathbf{A}+\mathbf{B}! \notin Z^+|\mathbf{A}+\mathbf{B}$.

Summary—The objects introduced above constitute a conceptual framework of the developing AI knowledge model for conflict management in a FFS. This definition system requires further refinement. Designing algorithms for this object system processing constitute a separate task. However, two important algorithmic issues are discussed below. These are (1) the use of Reynolds' simple steering behaviors in intended flight path planning for potential conflict avoidance, and (2) FSTN-based prediction of impending conflicts under demanding conditions.

7. Simple Steering Behaviors and Free Flight

In this section, the applicability of the Reynolds' model of a natural group motion, such as bird flocking, fish schooling, or insect swarming [5], [11]-[13], to the FF problem is discussed.

Similarities—There are several common features between FF and a flocking phenomenon. In both cases a six-degree-of–freedom motion of the participants is a result of action of aero- or hydrodynamic forces. An individual's control within the group is based on the imperative of 'no-collision' with the rest of the group and with external obstacles. Further, similar to the 'safety bubble' around a vehicle in the AI model, a bird probably has a kind of virtual wrap-around sphere for situational awareness. Based on results of its neighborhood scanning (by means of visual and other sensors), the bird probably 'computes' a required steering force for each control surface. These elementary computations constitute a multi-stage adaptive process. The distances to and the configuration of the bird's neighborhood determine the magnitude of the steering force and the required velocity. Unpredictable behavior of birds in a group emerges from the adaptive nature of individual behaviors [5].

Potential advantages—The self-organizing principles of flocking are demonstrated in online simulations at [11]-[13]. In particular, these experiments indicate that the computational complexity of 'traffic control' in a flock does not depend critically upon the flock size. *"Flocks do not become "full" or "overloaded" as new birds join. When herring migrate toward their spawning grounds, they run in schools extending as long as 17 miles and containing millions of fish"* [5]. Another important feature of flocking confirmed in simulations is that in a large flock, an individual participant has a localized and filtered perception of its surrounding environment [5]. Thus, the self-organizing behavior would be an ideal solution to the FF problem in close air space. Applications include coherent flight of a group of vehicles, such as a flow of 'flying cars' in urban areas, a cortege of autonomous cargo air transports following a leader, unmanned combat missions and other tasks.

Simple steering behaviors

There are nine basic navigation rules, which Reynolds calls 'simple steering behaviors'. These are [13]: Seek and Flee, Pursue and Evade, Wander, Arrival, Obstacle Avoidance, Containment, Wall Following, Path Following, and Flow Field Following. These rules may be used as 'building blocks' of the self-organizing collective behavior in a group of vehicles on a higher level. For example, this may include the following navigation tasks: get from one location to another while avoiding obstacles, go down a 'corridor', join a group of vehicles, follow a leader, help a 'blind' vehicle, loiter, or queue for landing, etc. It is essential that these tasks can be implemented without external supervision. The following is a brief discussion of the 'simple steering behaviors' proposed by Reynolds in conjunction with the FF problem based on [5], [11]-[13].

Seek and Flee—A vehicle seeks or flees from a 'target' located at a specified point in air space V. The behavior is accomplished when the vehicle gets close to the target or the distance exceeds some threshold (e.g.: $d_{\min}$). *Seek* attempts to steer a vehicle so that it moves toward the 'target'. A steering force vector is a function of the difference between the current velocity and the desired velocity (towards the target for *Seek*, away from the target for *Flee*).

Pursue and Evade—One vehicle pursues a "target" vehicle, while another vehicle tries to evade it. The future flight path of the "target" is estimated based on its position, velocity and distance to the pursuer (or evader). The first pursuing vehicle uses *Seek* to approach the estimated target position, while the evading vehicle uses *Flee* to escape from it.

Wander—A vehicle 'wanders' through a close air space. This is a type of random behavior, in which the steering direction on one step depends on the steering direction on the next step. At each time step a random increment is added to the wander direction. The modified wander direction is constrained by a sphere – ref. [13] for more detail.

Arrival—The goal of the vehicle is to move towards a 'target' located at a specified point in space. The vehicle attempts to arrive there with a specified (about zero or other) velocity. The vehicle is initially located at a certain distance from the target, moving in an arbitrary direction. Also specified is the maximum distance at which the vehicle may begin to slow down.

Obstacle Avoidance—A vehicle avoids obstacles, trying to remain outside of a prohibited zone. This behavioral pattern uses predictions of the vehicle's future path. Any obstacle that intersects these projections is treated as a potential collision. The nearest threat is chosen for immediate resolution. To avoid an obstacle, lateral and vertical control forces are applied. In addition, deceleration or acceleration can be used. However, simulation experiments demonstrate [13] that in a crowded environment the collisions with obstacles do occur under such tactics.

Containment—A vehicle uses *Containment* (or generalized obstacle avoidance) to stay off the volumes marked as prohibited zones. This behavior is related to *Obstacle Avoidance*. The obstacles can be of arbitrary shapes. *Containment* allows the vehicle to navigate close to the obstacle's surface. The vehicle checks the space ahead of it with probe points, which belong to the vehicle's intended flight path or, more precisely, to the surface of its safety corridor. When a probe point touches an obstacle, it is projected on to the nearest point on the obstacle surface, and the normal to the surface at that point is determined (ref. [13]). Communication between vehicles and an obstacle is conducted according to a generic surface protocol [13]. Again, this behavior needs no knowledge of the surface's shape.

Wall Following—This pattern is used to move parallel to and offset from prohibited zones called 'walls'. The vehicle's goal is to remain a given distance from the zone ('wall') as it moves. To implement *Wall Following,* prediction of the vehicle's position (intended flight path) is used. The vehicle next future position is projected to the nearest point on the 'wall'. Moving out from this point along the 'wall' normal by the desired offset distance produces a target point. Finally, *Seek* behavior is employed to steer toward the target point. Communication between the vehicle and the 'wall' is conducted by a generic protocol [13]. This steering behavior needs no knowledge of the prohibited zone's shape.

Path Following—The vehicle's task is to traverse the path in a specified direction while keeping its center within a certain corridor. The intended flight path - see the definition in the previous section - is exemplified by a series of connected line segments and a radius [13]. This behavior is related to *Containment* and *Wall Following,* but differs because the path has a direction. Corrective control inputs are applied only when the vehicle begins to deviate from the corridor. The vehicle's intended flight path is dynamically updated. When a corrective action is required (see the criterion in [13]), it is obtained using the *Seek* rule on a target point further down the flight path. As a result, this behavior needs no knowledge of the path's profile.

Flow Field Following—A vehicle attempts to align its motion with the local tangent of a flow (force, vector) field. The flow field defines a mapping from a location in space to a flow vector. However, as Reynolds points out [13], this behavior is sensitive to high spatial frequencies and discontinuities in the field, but these features will unlikely be characteristic to a FFS. The field may account for the vehicle individual mass, speed, as well as for the arrangement of $\Omega(\mathbf{V})$ in V. The field can be defined algorithmically (by a table of data points) or analytically. The field can be static or non-stationary.

Combined behaviors and groups

The Reynolds' simple steering behaviors can be combined to implement a higher level tasks of autonomous navigation of a group of vehicles, including the following [13]: *Crowd Path Following*, *Leader Following*, *Unaligned Collision Avoidance*, *Queuing* (at a doorway), and *Flocking* itself. Following is a FF interpretation of some of these patterns based on [13].

Crowd Path Following—A system of N vehicles implement a crowd path following behavior using the *Path Following* behavior. This behavior includes *Separation* to keep the vehicles from clumping together. In addition, the vehicles' motion is restricted by a kinematic non-penetration constraint to prevent them from colliding. However, the tactics used here is not physics-based.

Leader Following—A group of vehicles is trying to follow a 'leader'. The leader's behavior can be arbitrary and is not necessarily communicated to the followers. The *Leader Following* behavior combines *Separation* (to avoid crowding) and *Arrival*. The arrival target is a point offset behind the leader. In addition, a follower will have to move out of the way if finds itself on the near term future path of the leader.

Unaligned Collision Avoidance—Vehicles try to steer a collision-free path, threading through the group while attempting to stay away from a prohibited zone and environmental obstacles (*Containment*). Collision avoidance is employed for a vehicle by considering each of the other vehicles, and determining (based on current velocities) when and where the two will make their nearest approach. A potential for collision exists if that nearest approach is in the future, and if the distance between the vehicles (e.g. **A** and **B**) at nearest approach is insufficient, i.e. $(\mathbf{A})\cap(\mathbf{B}) \neq \varnothing$. The nearest of these potential collisions, if any, is determined. The vehicle then steers to avoid the point of the potential collision by applying control forces in required directions. It will also accelerate or decelerate to get to the site before or after the predicted collision. However, in computer simulations with this model each vehicle collides about 5 times per 1000 simulation steps [13].

Queuing—Vehicles may employ *Queuing* to leave an air space in a specified direction, e.g.: through a narrow 'doorway' [13]. *Queuing* results from a steering behavior of the vehicle, which decelerates when other vehicles are detected nearby, in front of, and moving slower. In addition, these vehicles are drawn toward the 'doorway' by *Seek* behavior; they avoid the prohibited zones and obstacles and maintain separation from each other. A kinematic inter-penetration constraint prevents them from overlapping with each other or the 'walls' (obstacles).

Summary—There is a similarity between the task of collision avoidance in a group of vehicles merging arbitrarily in a close air space and collective behavior in a flock of birds. FF in close air space can be represented as a decentralized, self-organizing process. In this process, collision avoidance decisions are based on kinematic and geometric properties of individual vehicles and their disposition; knowledge of the vehicle dynamics is not employed. The Reynolds' simple steering behaviors are suitable for developing rules for autonomous separation in an AI model of FF. Reynolds' simulations demonstrate that the chances of collisions still remain if a close air space is overcrowded and contains environmental obstacles.

8. Principles of Knowledge-Based Resolution of Impending Conflicts

To address the problem of impending conflict management under demanding conditions, it is suggested to use knowledge accumulated in the FSTN. This method can be useful in those situations where self-organizing tactics does not 'see' an about-to-happen conflict due to lack of knowledge of the system dynamics. In this section such a method will be introduced using a hypothetical FFS example.

Example—Let a group of four, heterogeneous in general, vehicles, $\Omega(\mathbf{V}) = \{\ \mathbf{A},\ \mathbf{B},\ \mathbf{C},\ \mathbf{D}\ \}$, merge arbitrarily in a close dynamic air space V_D, i.e. $d_{\mathbf{X},\mathbf{Y}} \leq d_{\min}$, $\mathbf{X}$, $\mathbf{Y} \in \Omega(\mathbf{V})$ and $d_{\mathbf{X},\mathbf{Y}} = f(t)$. The situation development is shown in Fig. 12, a-c. A series of 'snapshots' of the situation is made at three successive time instants t_0, t_1, and t_2, $t_i \in \Omega(t)$. These diagrams show the vehicles' intended flight paths and non-standard ('what-if') trajectories in projection on a local horizontal plane.

Assumptions—It is assumed that the intended flight paths for this group of vehicles are known (for example, from a self-organization model of Reynolds [13]). Also, there exist communication links between the vehicles based on the ADS-B or similar protocol. This allows each participant to receive data on the intended paths and relevant FSTN information from other vehicles, as well as to broadcast its own flight path and FSTN data.

FSTN-based prediction of future flight paths—This prediction process works as follows. At the current time instant t_i, $t_i \in \{\ \ldots;\ t_0;\ t_1;\ t_2;\ \ldots\ \}$, $t_i < t_*$, each vehicle **V** from $\Omega(\mathbf{V})$ 'scans' a FF sub-domain, $Z|\mathbf{V}$, ahead, both in time and in space, with the objective to identify impending conflicts. For this purpose, **V** uses its current state data $\boldsymbol{x}_{\mathbf{V}}(t)$ and information stored in the $\text{FSTN}_{\mathbf{V}}$. This sub-domain is basically a 'what-if' neighborhood of $\mathbf{V}_0$ constructed under key operational hypotheses. The number of these hypotheses is kept relatively small (within 10^2-10^4). The hypotheses normally account for actually present and anticipated operational conditions. They may reflect the pilot's desire to backup his (her) knowledge of the 'pilot - vehicle - operational environment' system dynamics under demanding conditions, which are characteristic to a particular vehicle type or phase of flight.

It is assumed that the pilots have selected the following sets of key hypotheses for examination: $\Omega(\mathsf{H}|\mathbf{A})$, $\Omega(\mathsf{H}|\mathbf{B})$, $\Omega(\mathsf{H}|\mathbf{C})$, and $\Omega(\mathsf{H}|\mathbf{D})$, where $\Omega(\mathsf{H}|\mathbf{V}) = \{ \mathsf{H}_{\mathbf{V}1}(\mathbf{\Phi}), \ldots, \mathsf{H}_{\mathbf{V}i}(\mathbf{\Phi}), \ldots, \mathsf{H}_{n(\mathbf{V}i)}(\mathbf{\Phi}) \}$, $\mathbf{V}\in\Omega(\mathbf{V})$. In our example, $n(\mathbf{A}_i) = n(\mathbf{B}_i) = n(\mathbf{D}_i) = 21$ and $n(\mathbf{C}_i) = 20$. After 'joining' the FFS, each new vehicle $\mathbf{V}$ examines its possible ('what-if') fuzzy flight paths to check these hypotheses. For this purpose, a sub-tree of fuzzy path-branches $\{ \mathbf{V}_1, \ldots, \mathbf{V}_i, \ldots, \mathbf{V}_{n(\mathbf{V}i)} \}$ from the $\mathrm{FSTN}_\mathbf{V}$ is projected into V_D forming a probe cone ahead of $\mathbf{V}$. This cone wraps up the intended path $\mathbf{V}_0$ and is translated along it as the vehicle moves in V_D. The objectives of this process are to: (1) detect zones in close air space, where fuzzy flight paths for two or more vehicles intersect under key operational hypotheses, (2) analyze these zones quantitatively, and (3) find fuzzy paths and recovery tactics to avoid these zones. Note that the depth of FSTN-based flight path predictions is $[t_*; t^*]$, where $t_* > t_i$.

Following is a description of this multi-stage process of conflict management at t_0, t_1, and t_2 based on the hypothetical example described above.

Situation at t_0—The vehicles at the initial time instant t_0 are depicted in Fig. 12, a, together with their intended flight paths $\{ \mathbf{A}_0, \mathbf{B}_0, \mathbf{C}_0, \mathbf{D}_0 \}$ and relevant sub-trees from $\mathrm{FSTN}_\mathbf{A}$, …, $\mathrm{FSTN}_\mathbf{D}$. It follows from the diagram that no impending collisions are identified at t_0. That is, none of the FSTN crowns projected at t_0 'touches' other ones within the prediction range $[t_*; t^*]$, i.e. $Z|\mathbf{A} \cap Z|\mathbf{B} \cap Z|\mathbf{C} \cap Z|\mathbf{D} = \varnothing$. As a result, no warning or resolution advisory is generated at t_0.

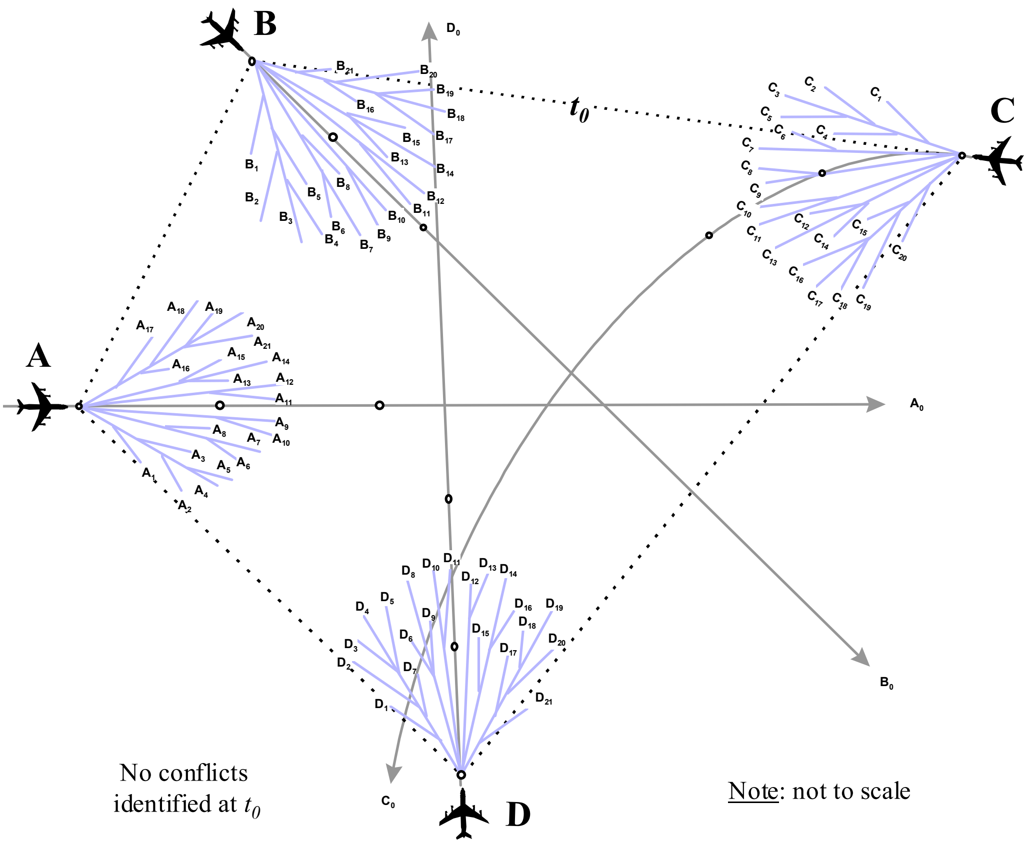


Figure 12, a: Free flight situation 'snapshot' at time t_0

Situation at t_1—At the next time t_1 (Fig. 12, b), $t_1 = t_0 + \delta$, five impending conflicts are identified in pairs ($\mathbf{A}$, $\mathbf{B}$) and ($\mathbf{A}$, $\mathbf{D}$) at times $t_{1!}$, $t_{2!}$, $t_{3!}$, $t_{4!}$, and $t_{5!}$, where $t_{k!} \in [t_*; t^*]$ and $t_1 > t_*$. In the example, these conflict subsets are $\Omega(\mathbf{A+B!})$ and $\Omega(\mathbf{A+D!})$, where $\Omega(\mathbf{A+B!}) = \{ \mathbf{A}_{18}+\mathbf{B}_2|t_{1!}; \mathbf{A}_{20}+\mathbf{B}_3|t_{2!} \}$ and $\Omega(\mathbf{A+D!}) = \{ \mathbf{A}_4+\mathbf{D}_4|t_{3!}; \mathbf{A}_5+\mathbf{D}_5|t_{4!}; \mathbf{A}_7+\mathbf{D}_8|t_{5!} \}$.

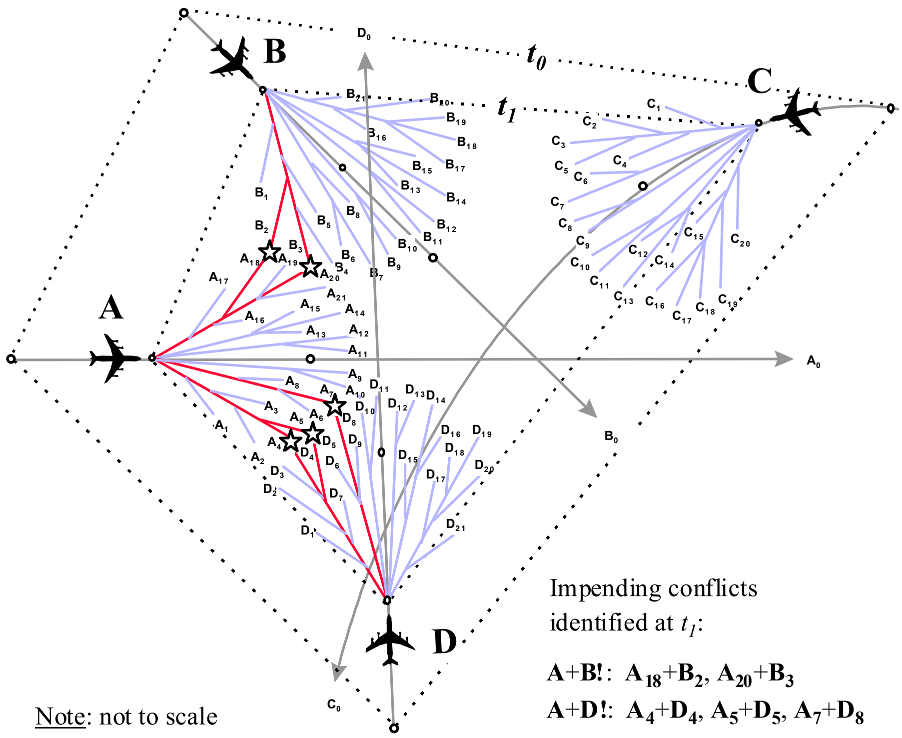


Figure 12, b: Free flight situation 'snapshot' at time t_1

First, the united set of conflicts, $\Omega(\mathbf{A+B!}) \cup \Omega(\mathbf{A+D!})$, is sorted in the ascending order of the time instants { $t_{1!}$, $t_{2!}$, $t_{3!}$, $t_{4!}$, $t_{5!}$ }. The possibility for conflicts is verified within the affected pairs, (**A**, **B**) and (**A**, **D**). After that, this information is announced to the rest of the group. In the announcement, two zones in V_D are declared as prohibited: $Z|\mathbf{A+B}(t_1)$ and $Z|\mathbf{A+D}(t_1)$. Also, a subset of interrelated operational hypotheses, $\Omega(\mathrm{H}|t_1)$, under which the affected vehicles can be brought to zones Z, is formed, i.e. $\Omega(\mathrm{H}|t_1) = \{ \mathrm{H}_{\mathbf{A18}}(\Phi)+\mathrm{H}_{\mathbf{B2}}(\Phi), \mathrm{H}_{\mathbf{A20}}(\Phi)+\mathrm{H}_{\mathbf{B3}}(\Phi), \mathrm{H}_{\mathbf{A4}}(\Phi)+(\mathrm{H}_{\mathbf{D4}}(\Phi), \mathrm{H}_{\mathbf{A5}}(\Phi)+\mathrm{H}_{\mathbf{D5}}(\Phi), \mathrm{H}_{\mathbf{A7}}(\Phi)+\mathrm{H}_{\mathbf{D8}}(\Phi) \}$. These pairs are placed on the top of the 'check list' of health and weather monitoring systems for the vehicles **A**, **B** and **D**.

In addition, a joint recovery tactics, $\mathrm{T}_{\mathbf{A}\Uparrow}(\Phi)+\mathrm{T}_{\mathbf{B}\Uparrow}(\Phi)$, is searched in $\mathrm{FSTN}_{\mathbf{A}}$ and $\mathrm{FSTN}_{\mathbf{B}}$, if: (1) a pair of hypotheses, e.g.: $\mathrm{H}_{\mathbf{A18}}(\Phi)+\mathrm{H}_{\mathbf{B2}}(\Phi)$, becomes actual, (2) vehicles **A** and **B** are still on or close to the conflicting branches, and (3) the chances of recovery are lower than some safety margin [6]. If the conflict threat persists and a decision is made to recover, this tactics can be applied to **A** and **B** according to the flight control scenarios stored in the $\mathrm{FSTN}_{\mathbf{A}}$ and $\mathrm{FSTN}_{\mathbf{B}}$ together with other information on the paths $\mathbf{A}_{\Uparrow}$ and $\mathbf{B}_{\Uparrow}$. The same principles can be applied to construct a joint recovery tactics for the vehicles **A** and **D**.

Situation at t_2—If no decision to recover has been made at t_1, a new cycle of flight path prediction is initiated at t_2. Due to a high density and overlapping of flight path-branches for several vehicles, it would be difficult to comprehend the situation from a single diagram. To ease the process, a pairwise analysis of impending conflicts at t_2 is shown on a scaled diagram in Fig. 12, c (see also a legend to this diagram in Fig. 12, d). It follows from Fig. 12, c, that six subsets of impending conflicts have been identified at t_2: **A+B!**, **A+C!**, **A+D!**, **B+C!**, **B+D!**, and **C+D!**. These subsets are defined below: $\Omega(\mathbf{A+B!}) = \{ \mathbf{A}_{11}+\mathbf{B}_9, \mathbf{A}_{14}+\mathbf{B}_{10}, \mathbf{A}_{16}+\mathbf{B}_3, \mathbf{A}_{17}+\mathbf{B}_2, \mathbf{A}_{18}+\mathbf{B}_5, \mathbf{A}_{21}+\mathbf{B}_{11} \}$, $\Omega(\mathbf{A+C!}) = \{ \mathbf{A}_9+\mathbf{C}_{10}, \mathbf{A}_{12}+\mathbf{C}_8, \mathbf{A}_{14}+\mathbf{C}_7 \}$, $\Omega(\mathbf{A+D!}) = \{ \mathbf{A}_{13}+\mathbf{D}_{17}, \mathbf{A}_{16}+\mathbf{D}_{12}, \mathbf{A}_{20}+\mathbf{D}_{16}, \mathbf{A}_{21}+\mathbf{D}_{19} \}$, $\Omega(\mathbf{B+C!}) = \{ \mathbf{B}_{12}+\mathbf{C}_7, \mathbf{B}_{14}+\mathbf{C}_5, \mathbf{B}_{15}+\mathbf{C}_3, \mathbf{B}_{18}+\mathbf{C}_2, \mathbf{B}_{20}+\mathbf{C}_1 \}$, $\Omega(\mathbf{B+D!}) = \{ \mathbf{B}_1+\mathbf{D}_{10}, \mathbf{B}_3+\mathbf{D}_{12}, \mathbf{B}_4+\mathbf{D}_{18}, \mathbf{B}_9+\mathbf{D}_{20} \}$, and $\Omega(\mathbf{C+D!}) = \{ \mathbf{C}_7+\mathbf{D}_{20}, \mathbf{C}_9+\mathbf{D}_{21} \}$.

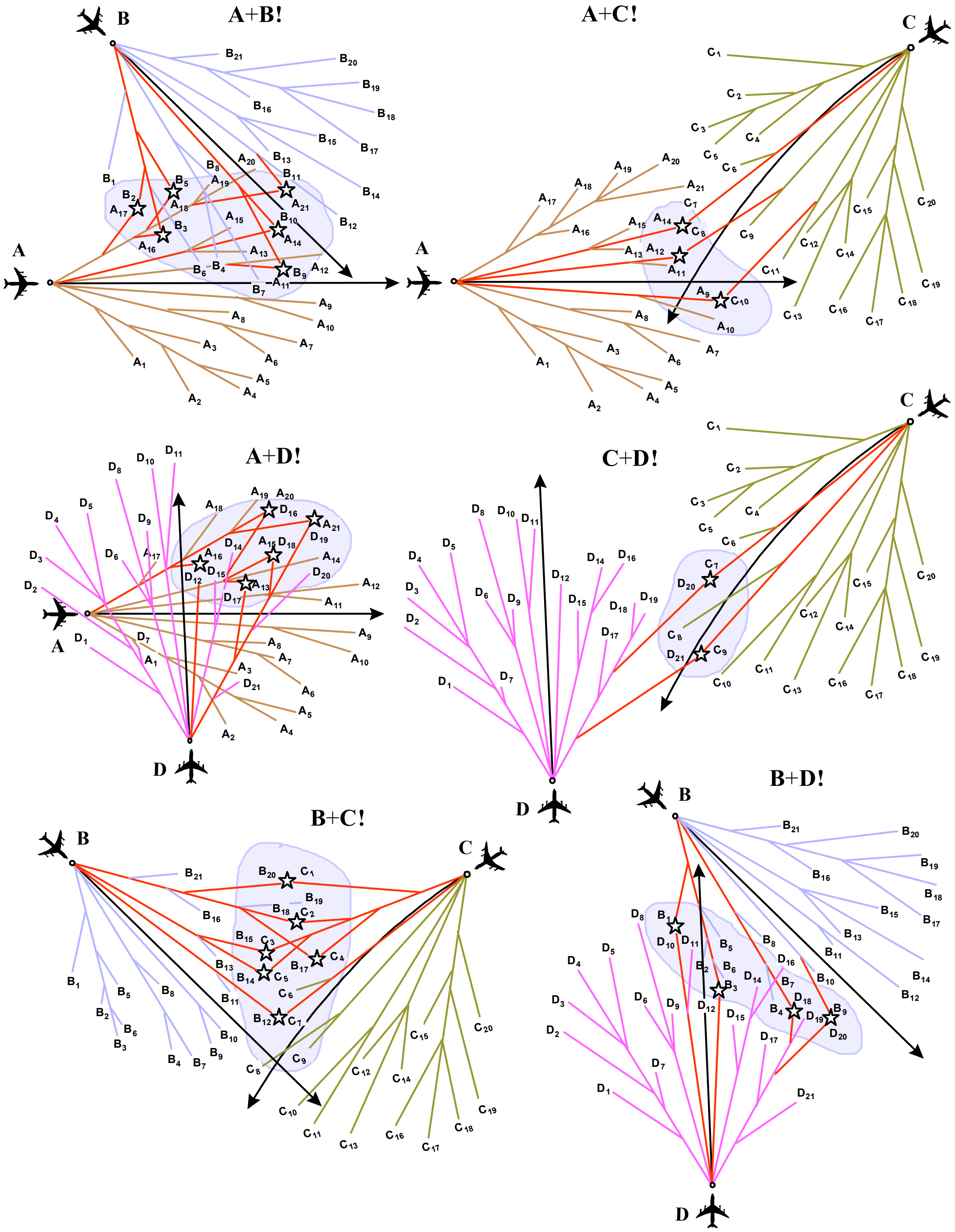


Figure 12, c: Pairwise analysis of impending conflicts at time $\boldsymbol{t_2}$

All impending conflicts identified at t_2 (24 in total) may have different occurrence times within $[t_*; t^*]$. By applying urgency and criticality criteria, these conflicts can be prioritized for resolution. For example, most critical hypotheses, which include uncontrollable and other strong factors, must be accounted first. Note also that none of the impending conflicts from the previous step repeats at t_2. This may happen because the situation is essentially dynamic.

A joint recovery tactics is then searched for the affected vehicles to address the most urgent and critical impending conflicts (see the criterion described on the previous step). Also, one more requirement is to be added. The zone of allowed recovery maneuvers, $\mathcal{Z}^{+}|\mathbf{X}+\mathbf{Y}$, in this particular case (at t_2) must also obey the following 'no-intersection' condition: $(\forall \mathbf{X}, \mathbf{Y})$ $(\mathbf{X}, \mathbf{Y} \in \Omega(\mathbf{V})$ and $\mathbf{X} \neq \mathbf{Y})$ $(\mathcal{Z}^{+}|\mathbf{X}+\mathbf{Y} \not\subset \mathcal{Z}|\mathbf{A}+\mathbf{B!} \cup \mathcal{Z}|\mathbf{A}+\mathbf{C!} \cup \mathcal{Z}|\mathbf{A}+\mathbf{D!} \cup \mathcal{Z}|\mathbf{B}+\mathbf{C!} \cup \mathcal{Z}|\mathbf{B}+\mathbf{D!} \cup \mathcal{Z}|\mathbf{C}+\mathbf{D!})$. This condition means that no intersection is permitted between any of the allowed recovery maneuvers zones and any of the prohibited zones. In Fig. 12, c these prohibited zones are shadowed.

Pilot-vehicle interface—Knowledge of impending conflicts can be conveyed to the pilot, for example, in the form of a Situational Forecast Display (SFD) [6], [10]. This notional format can be adapted to account for a multiple vehicle system similar to that one exemplified in Fig. 12, c.

X

1. $\mathbf{X}_0$ - intended (safe) flight path for vehicle **X**
2. $\mathbf{X}_i$ - non-standard branch $\mathbf{X}_i$ for vehicle **X**
3. $\mathbf{Y}_j$ $\mathbf{Y}_j$ - impending conflict between vehicle **X** and vehicle **Y**, i.e. $\mathbf{X}_i+\mathbf{Y}_j$
4. - zone of impending conflicts for $[t_*; t^*]$
5. List of impending conflicts in group { **A**, **B**, **C**, **D** }:

A+B!: $\mathbf{A}_{11}+\mathbf{B}_9$, $\mathbf{A}_{14}+\mathbf{B}_{10}$, $\mathbf{A}_{16}+\mathbf{B}_3$, $\mathbf{A}_{17}+\mathbf{B}_2$, $\mathbf{A}_{18}+\mathbf{B}_5$, $\mathbf{A}_{21}+\mathbf{B}_{11}$
A+C!: $\mathbf{A}_9+\mathbf{C}_{10}$, $\mathbf{A}_{12}+\mathbf{C}_8$, $\mathbf{A}_{14}+\mathbf{C}_7$
A+D!: $\mathbf{A}_{13}+\mathbf{D}_{17}$, $\mathbf{A}_{16}+\mathbf{D}_{12}$, $\mathbf{A}_{20}+\mathbf{D}_{16}$, $\mathbf{A}_{21}+\mathbf{D}_{19}$
B+C!: $\mathbf{B}_{12}+\mathbf{C}_7$, $\mathbf{B}_{14}+\mathbf{C}_5$, $\mathbf{B}_{15}+\mathbf{C}_3$, $\mathbf{B}_{18}+\mathbf{C}_2$, $\mathbf{B}_{20}+\mathbf{C}_1$
B+D!: $\mathbf{B}_1+\mathbf{D}_{10}$, $\mathbf{B}_3+\mathbf{D}_{12}$, $\mathbf{B}_4+\mathbf{D}_{18}$, $\mathbf{B}_9+\mathbf{D}_{20}$
C+D!: $\mathbf{C}_7+\mathbf{D}_{20}$, $\mathbf{C}_9+\mathbf{D}_{21}$

Figure 12, d: Legend to **Fig. 12, c** "Pairwise analysis of impending conflicts at time t_2"

Summary—The objects formalized in the definition section were applied to formalize a FFS using a hypothetical example. This example illustrates how the concept of fuzzy situational tree-network can be applied to implement and automate the process of conflict management. One of the problems, which require further study, is communication of conflict related knowledge within a group of conflicting vehicles.

9. CONCLUSION

A conceptual system of an AI knowledge model for conflict management in close free flight air space has been developed. The model incorporates natural principles of self-organization of collective behavior (flocking, swarming, etc.) discovered by C. Reynolds. These principles can be used to manage potential conflicts in a close air space based on kinematic and geometric constraints. The model includes a knowledge base of complex system dynamics in the form of fuzzy situational tree-network (FSTN) of flight. The FSTN purpose is to predict and resolve impending conflicts under uncertain and multi-factor conditions. Combination of self-organization and physics-based prediction of flight is essential, because a flocking model alone may not be sufficient to avoid collisions in crowded air space. Collision threats may suddenly emerge due to an unforeseen action of several demanding factors.

The purpose of the FSTN is to thread close air space around the vehicle's intended flight path. These alternative fuzzy path-branches can be specially programmed to account for critical operational hypotheses and multi-factor situations. Thus, impending conflicts can be predicted dynamically as a result of projection and overlap probing of fuzzy situational sub-trees for affected vehicles 5-15 seconds ahead. A flight path conflict is identified if fuzzy path-branches for two or more vehicles intersect in the near term future. These sub-trees also contain information on the available safe paths, which can be employed to resolve conflicts.

The developed model belongs to a class of memory-based distributed artificial intelligence with the capability of self-organization and knowledge-based near term prediction of flight paths. Its purpose is to serve as a knowledge backup for the pilot or an automatic recovery system in congested FF air spaces. External supervision or control is not required.

Future research is expedient to continue in the following directions:
- refine the model's conceptual framework; incorporate self-organization principles into the framework
- adapt Reynolds' simulation experiments in bird flocking to a group of heterogeneous flight vehicles using 6DOF non-linear flight dynamics models
- develop a FSTN genotype for FF applications; construct and test this prototype
- design 'intelligent formats' for representing knowledge of a complex FF situation to the pilot
- test the model in autonomous and manned simulations.

REFERENCES

[1] RTCA, "Report of the Radio Technical Commission for Aeronautics (RTCA) Board of Directors' Select Committee on Free Flight", Washington, DC, RTCA, 1995

[2] Van Gent, R.N.H.W, Hoekstra, J.M., and Ruigrok, R.C.J. "Free Flight With Airborne Separation Assurance", *Proceedings of the International Conference on Human-Computer Interaction in Aeronautics (HCI-Aero'98), Montreal, Quebec, May 1998*, pp. 63-69

[3] Goldin, D.S. "The Three Pillars of Success for Aviation and Space Transportation in the 21st Century", *A Speech before the Aero Club, AIAA, and NAC, Thursday, 20 March 1997*

[4] Parasuraman, R., Duley, J., Galster, S., Masalonis, A., Hilburn, B. "Dynamics Automation Tools to Support Air Traffic Controllers Under Free Flight", *Proc. of the International Conference on Human-Computer Interaction in Aeronautics (HCI-Aero'98), Montreal, Quebec, May 1998*, pp. 71-76

[5] Reynolds, C.W. "Flocks, Herds, and Schools: A Distributed Behavioral Model", *Computer Graphics,* **21**(4), Proceedings of the SIGGRAPH '87 Conference, pp. 25-34, 1987

[6] Burdun, I.Y. "An AI Situational Pilot Model for Real-Time Applications", *Proceedings of the 20th Congress of the International Council of the Aeronautical Sciences, Sorrento, Napoli, Italy, 8-13 September 1996 (ICAS'96)*, Volume 1, AIAA, 1996, pp. 210-237

[7] Burdun, I.Y. and Mavris, D.N. "A Technique for Testing and Evaluation of Aircraft Flight Performance In Early Design Phases" (Paper No. 975541), *Proceedings of the World Aviation Congress (WAC'97), Anaheim, October 1997,* AIAA, 13 pages

[8] Burdun, I.Y., Parfentyev, O.M. "Fuzzy Situational Tree-Networks for Automated Flight Envelope Protection and Pilot-Vehicle Intelligent Interface*", Proceedings of the Workshop on Artificial Intelligence for Aeronautics and Space (ICTAI'96), November 16, 1996,* ONERA-CERT, Toulouse, France, 1996, pp. 15-22

[9] Burdun, I.Y., and Parfentyev, O.M. "Analysis and Optimization Of Hypersonic Maneuvering Of A Transatmospheric Vehicle Under Uncertainty Using Fuzzy Trees" (Paper No. AIAA-98-4976), *Seventh AIAA/ USAF/ NASA/ ISSMO Symposium on Multi-disciplinary Analysis and Optimization, September 2-4, 1998*, St. Louis, 10 pages

[10] Burdun, I.Y. "The Intelligent Situational Awareness And Forecasting Environment (The S.A.F.E. Concept): A Case Study" (Paper No. 981223), *Proceedings of the SAE Advances in Flight Safety Conference and Exhibition, April 6-8, 1998, Daytona Beach, FL, USA" (P-321)*, AIAA - SAE Aerospace, 1998, pp. 131-144

[11] Reynolds, C. "Individual-Based Models. An Annotated List of Links", http://hmt.com/ cwr/ ibm.html

[12] Reynolds, C. "Boids. Background and Update", http:// hmt.com/cwr/boids.html

[13] Reynolds, C. "Steering Behaviors For Autonomous Characters", http://hmt.com/cwr/steer/

[14] Wirth, N. "Programming in Modula-2", Springer-Verlag, 1988, page 87

[15] Burdun, I.Y. "Flight Certification and Safety by Design. Introduction of Virtual Testing and Certification." Lecture Course (AE 8133), *Lecture notes; XV-15, HSCT modeling and simulation project handouts*, School of AE, Georgia Tech, Spring Quarter 1998

[16] Schrage, D.P., Calise, A.J., Burdun, I.Y., Pritchett, A., and Rysdyk, R.T. "An Integrated Knowledge-Based Approach to Improving Aircraft Safety" (White Paper), School of Aerospace Engineering, Georgia Institute of Technology, October 1998, 20 pages

[17] Korn, G.A., and Korn, T.M. "Mathematical Handbook for Scientists and Engineers", McGraw-Hill, 1968

[18] Macaulay, W.H. "Solid Geometry", Cambridge University Press, 1930

[19] Jeger, M., Eckmann, B. "Vector Geometry and Linear Algebra", John Wiley & Sons, 1967

[20] Alterman, S. "The Transition to ADS-B", *Traffic World*, January 5, 1998

[21] "Proposed ME-Field Format Definitions For ADS-B Extended Squitter Messages", RTCA/SC-186/WG-2, 1090 MHz MOPS Working Group, Melbourne, Florida, 17-19 March 1998 [http://www.iimorrow. com/ caa_ads/ adsb-info/ modes/56bits.html]

BIOGRAPHIES

Ivan Y. Burdun—Ivan graduated as Aeronautical Mechanical Engineer from the Riga Civil Aviation Engineers Institute (RCAEI), ex-USSR. His PhD thesis (1982, RCAEI) was with fuzzy sets based computer simulation of aircraft flight dynamics and human piloting under complex conditions. In 1983-1992 Dr. Burdun worked as Senior Researcher at Flight Modeling Department of the Riga Experimental Center for Aeronautical Research of the USSR State Research Institute of Civil Aviation (GosNII GA). He had carried out over 20 research contracts for several aerospace customers as Principal

Investigator and Co-Investigator. In 1993-96 he had received a second PhD training in Avionics and AI under an ORS Award at Cranfield University, UK (writing up). He has implemented autonomous situational models of flight for 19 aircraft types. Using these models, over 30 practical problems were studied in the sectors of practical aerodynamics, flight control, and flight safety. He has authorized over 80 technical reports and other publications in Russian and English, three applied techniques, one invention, and many Russian-English aviation translations. His research interests include hybrid intelligence pilot models, fuzzy situational decision trees, computer simulation of complex dynamics of the 'pilot - vehicle - operational environment' system, flight accident reconstruction and neighborhood analysis, aircraft virtual testing under multiple conditions, and onboard AI technologies for flight safety. At present Dr. Burdun works in the Aerospace Systems Design Laboratory, School of Aerospace Engineering, Georgia Institute of Technology. His responsibilities at Georgia Tech include research and teaching in the sectors of mathematical modeling, computer simulation and artificial intelligence for flight safety.

Oleg M. Parfentyev—Oleg graduated as Avionics Engineer from the Riga Civil Aviation Engineers Institute (RCAEI), ex-USSR. In 1993 he received a PhD degree in Flight Control and Cybernetics from the Department of Aircraft Automation of RCAEI. He worked as a Research Assistant and Research Engineer with Departments of Applied Mathematics and Aerodynamics, RCAEI, and Department of Avionics, the Riga Air Force Engineering College (1984-1992). Dr. Parfentyev participated in research and development projects for several major aerospace customers in the ex-USSR, including the State Research Institute of Civil Aviation, Molniya Scientific-Production Consortium, Center for Cosmonaut Training, Scientific-Research Institute of Space Medicine, and Air Force Engineering Academy, at the level of Co-Investigator and Principal Investigator. He has authorized over 30 technical reports, papers and conference presentations. His current research interests include AI flight control, pilot-vehicle anthropomorphic interface, fuzzy situational decision trees, and advanced pilot training. He has piloting experience with several Russian and Western aircraft types (over 1,700 flight hours, 12 airplanes, a helicopter, and 8 gliders). He also gained flight test experience with several home-built airplanes. At present, he is a flying instructor in sport aerobatics and soaring. He won several first and second prices in team competitions in soaring. Dr. Parfentyev is affiliated with the Siberian Aeronautical Research Institute (SibNIA, Russia) as Scientific Advisor to Chairman.