UDC 681.5.001.63:519.711(045)
DOI:10.18372/1990-5548.73.17013

[1]V. M. Sineglazov,
[2]S. Y. Savchuk

PLANNING A FLIGHT TASK SYSTEM AND THE PRINCIPLE OF ITS CONSTRUCTION

Aviation Computer-Integrated Complexes Department, Faculty of Air Navigation, Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine
E-mails: [1]svm@nau.edu.ua ORCID 0000-0002-3297-9060, [2]patcher_phoenix_0k@icloud.com

Abstract—*This work is devoted to the development of a system for automatic determination of the flight task for unmanned aerial vehicles. To implement such a system, the QT framework (a library of C++ classes and a set of tools for creating cross-platform applications) was used. It is proposed to introduce forecasting and routing modules into the system of automated determination of the flight task. An interface is developed for building a flight task that can be used to download, track key metrics from the unmanned aerial vehicles, and adjust the mission of the flight task. Advanced QT/Qml frameworks were used, which will allow the software product to be used on different operating systems, which will add flexibility in the use of system components.*

Index Terms—Unmanned aerial vehicle; Kalman method; Dijkstra algorithm; user interface; satellite image processing.

I. INTRODUCTION

The impetus for the development of unmanned aviation around the world was the need for light, relatively cheap aircraft with high maneuverability characteristics and capable of performing a wide range of tasks. Unmanned aerial vehicles (UAVs) are successfully used in military operations around the world, and at the same time they successfully perform civilian tasks.

The rapid development of unmanned aerial vehicle technologies in the world, as well as the growing demand for their technologies in the civil and commercial spheres of the world and Ukraine require additional research on the possibilities of their dual use development of the civilian market. To identify potential market opportunities technologies of unmanned aerial surveillance devices on the civilian market research is being conducted to expand the range of work that can be performed to be sought after by a certain group of market consumers and ego segments, evaluation consequences and risk during implementation.

Trends in the development and improvement of unmanned aircraft technology are closely related are connected with the continuation of the processes of structural restructuring of the industry, national priorities for the development of science and technology, the world market situation. The prerequisites for these changes are globalization of the economy, merger processes, interrelationships in the industry, development of information technologies.

Currently, there is an actual problem of safe integration unmanned aerial vehicles into airspace. Stakeholders conduct comprehensive examinations to create protective detection systems and warnings against unintended or illegal interference by drones c frequency sectors, to ensure echeloning relative to other aircraft, to develop a reliable regulatory framework based on the application of Standards of recommended practice (SORP), which are supplemented by air navigation rules service (ANRS), and also conducts medical examinations of crew members UAVs and issue certificates to them.

In this research, we study in detail the characteristics of UAVs, the purpose of their systems, and also develop a user interface for building a flight task using methods for building a route with satellite images included.

II. THE MAIN PRINCIPLES OF THE FORMATION OF THE FLIGHT TASK

The UAV autonomous flight scheme includes three main stages: flight task planning, route planning and autonomous flight using the control system. Flight route planning is understood as the search for the optimal route from its known initial position S_0 (starting point) to a given final position SF (destination), taking into account the dynamic characteristics of the UAV and solving the obstacle avoidance problem.

The flight route in real time is calculated taking into account the minimization of a certain indicator (flight time, fuel consumed, etc.). In order to solve this problem, an approach is proposed, the essence of which is as follows: it is necessary to develop a linearized dynamic model of the UAV; use as the main component, minimizing the objective function, the flight time between adjacent reference points, in which the flight route changes; apply partial-integer linear programming to introduce linear constraints with mixed forms, consisting of logical and continuous variables, to describe obstacles when avoiding obstacles.

If the complete flight route from the start point to the target point is calculated once, then the amount of computation will be quite large. At the same time, the calculation process is limited by the capabilities of the on-board computer: the allowable computation time and the amount of memory. On the other hand, if the information about the environment where the flight is performed is incomplete, then it is necessary to carry out its further research and clarification. Therefore, for an unknown environment, it is almost impossible to calculate the complete flight route once. When expanding or identifying new flight targets, the flight route can be formed and calculated for individual sections.

The need for planning a flight task is that this stage helps:

- to carry out intelligent multi-criteria planning of tasks under conditions of high uncertainty;
- calculate UAV routes taking into account the terrain and objects of the tactical situation;
- to automatically prepare flight tasks for autonomous UAVs;
- determine the order of joint actions and form flight tasks for groups of jointly operating UAVs in accordance with the quality criteria specified by the user (safety, assurance of target servicing, optimal use of available resources and equipment, etc.);
- adjust the assigned flight tasks in the course of their performance, taking into account the current set of technical means and their characteristics in real time;
- receive from external systems or set and change manually by the operator the initial data (on the tactical situation) for operational planning and control;
- simulate the processes of completing tasks to analyze the speed and quality of their implementation and calculate the need for resources (UAVs, equipment).

III. The Structure of the Formation of the Flight Task

The aim of the study is to create a program for constructing UAV flight tasks. The process of creating a program can be divided into the following functional stages.

1) Using the interface to create a UAV flight task.

2) Handling program events (program reactions to user interaction with the interface).

3) Setting up data transfer via communication channels.

4) Loading the flight task on the UAV.

5) Feedback from the UAV to the station.

To build a flight task, you need to use the right side of the application, the operator sets the initial take-off position of the UAV and, using the graphical interface, begins to set points on the map, guided by satellite images that were previously uploaded to the server. By double clicking on the map, he can set a waypoint and its coordinates appear in the top right of the window, if possible, they can also be changed by dragging one of them. After double clicking, the position of this point relative to the map area available on the screen is compared with the coordinates of the earth's surface and saved into the model for further processing (Fig. 1).

```
MouseArea{
    id: mainMouseArea
    anchors.fill:parent
    onDoubleClicked: {
        var inputCoordinate = myMap.toCoordinate(Qt.point(mouse.x,mouse.y))
        modelWaitPoint.append( { "latitude" : inputCoordinate.latitude,
                                 "longitude" : inputCoordinate.longitude,
                                 "activeState" : false } )
        polyLine.addCoordinate(inputCoordinate)
        mIndex++
    }
}
```

Fig. 1. Writing a correction point to the mode

All interactions in the program work thanks to signals that either signal some kind of action or transmit data to another area of the program where they are used for calculations or algorithms for performing a flight task A server is used to simulate a real UAV flight and transfer data to it. and a simple mathematical model, which contains the logic of a real lethal machine First, we send the data received using the interface to a separate class where they are structured and transmitted over a secure channel to device, an example of the structure can be seen in Fig. 2.

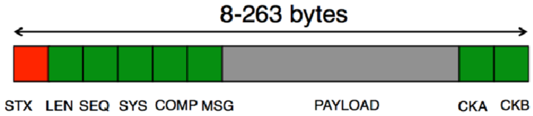

Fig. 2. Mission message structure

The protocol describes the information interaction between the UAV and the ground control station, as well as their constituent parts – components.

The first byte of the packet (STX) is the start character of the message; LEN is the payload (message) length. SEQ – contains a packet counter (0–255), which will help us identify the loss of a message. SYS (System ID) is the ID of the sending system, and COMP (Component ID) is the ID of the sending component. MSG (Message ID) – message type, it depends on what data will be in the payload of the package. PAYLOAD – packet payload, message, size from 0 to 255 bytes. The last two bytes of the packet – CKA and CKB, the lower and upper byte, respectively, contain the checksum of the packet.

The loading chart for mission elements shown in Fig. 3 consists of the following sequence:

1) The ground station makes a missionCount request to the MCU. The request defines the number of mission elements to be loaded (parameter missionCount). After sending the message, the NSO starts a timeout to wait for a response from the MBA.

2) The MBA receives the message and responds with a missionRequest message requesting the first element of the mission (sequence == 0). Drone timeout to wait for response message from missionItem ground station.

3) The ground station receives the missionRequest message and responds with the requested mission item with a missionItem message.

4) The MBA and the ground one repeat the missionRequest / missionItem cycle, iterating the sequence until all items are loaded (sequence == count – 1).

5) After receiving the last mission element, the MBA replies to the NSO with a missionResponse message with the type field equal to missionAccepted, which corresponds to the successful loading of all mission elements. (The MBA must set the new mission as the current mission by deleting the original data; the MBA considers the download complete).

6) The ground station receives a missionResponse with missionAccepted to indicate that the operation has been completed.

After the transfer of the flight task, we can track the key parameters from the UAV, for this, a special request is submitted from the ground station, which contains the information that we need to receive, in this case, these are the parameters of the object. The required request is sewn into the message structure (Fig. 4) and transmitted to the aircraft according to the protocol.

The final result is the display of all parameters and their visualization on the dashboard (Fig. 5).

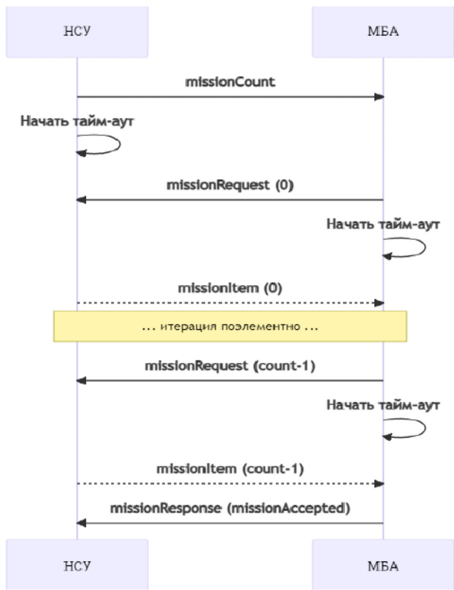

Fig. 3. Loading chart for mission elements

```
void Parser::slotSendMsg(StrMainPackage mainPckg, StrParamRequestList msg)
{
    QJsonObject paramRequestListObj =
            QJsonDocument::fromJson(paramRequestList.toUtf8()).object();

    paramRequestListObj["targetSystem"]    = msg.targetSystem;
    paramRequestListObj["targetComponent"] = msg.targetComponent;

    QJsonObject mainPckgObj{convertMainPackage(mainPckg)};

    mainPckgObj.insert("payload",paramRequestListObj);

    emit sendData(QJsonDocument(mainPckgObj));
}
```

Fig. 4. Method for requesting flight task parameters

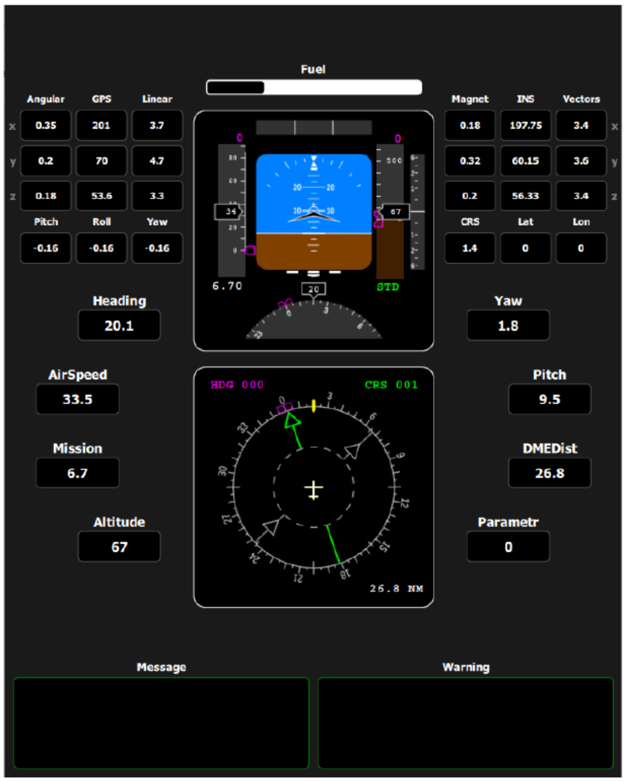

Fig. 5. UAV parameters display interface

IV. Planning and Management. General Information

This subsection provides a general overview of the architecture of the planning and control modules, as well as their submodules.

As shown in Fig. 6, the map and location module collects raw data such as point clouds and GPS data. Then it converts them into UAV location information. The Perception module is responsible for detecting objects in the immediate vicinity of the UAV. Both of these modules are focused on the perception of the objective world, while other modules, such as the module of routing, motion prediction, behavioral decision making, motion planning and feedback control, are focused on the subjective aspect: they are responsible for how the UAV predicts the behavior of the external environment, as well as its further movement in space.

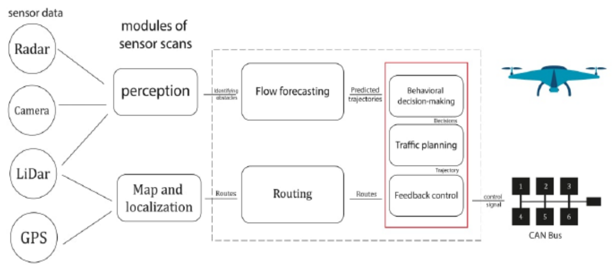

Fig. 6. Planning and control modules

All modules in Fig. 6 use a common central clock generator. In a clock cycle (also called a frame), each module independently retrieves the most recently published data from upstream modules, then performs its calculations and then publishes the result for use by subsequent modules.

V. Conclusions

In this work we have considered prediction and routing modules, neither of which are part of the traditional concept of planning and control modules. However, with our proposed broader understanding of the planning and control structure, forecasting and routing create inputs for traditional traffic planning. Therefore, they are included in the broader planning and control structure. Traffic forecasting is abstracted to a two-level classification of behavior and trajectory generation problems. Our proposed approach is path-level routing, according to which the UAV follows the UAV to its destination through sequences of defined trajectories. In the course of writing the work, an interface was developed to build a flight task that can be used to load, track key metrics from the UAV, and adjust the missions of the flight task.

References

[1] B. Paden, M. Cap, S. Z. Yong, D. Yershow, and E. Frazzolo, “A survey of motion planning and control techniques for self-driving urban vehicles,” *IEEE Transactions on Intelligent Vehicles*, 1(1), 2016, p. 33–55. https://doi.org/10.1109/TIV.2016.2578706

[2] M. Buehler, K. Lagnemma, and S. Sanjiv, (eds.). “The DARPA Urban Chal-lenge: Autonomous Vehicles in City Traffic,” *Springer Tracts in Advanced Ro-botics*. 2009. https://doi.org/10.1007/978-3-642-03991-1

[3] H. Sak, A. Senior, and F. Beaufays, “Long short-term memory recurrent neural network architectures for large scale acoustic modeling,” *In Fifteenth Annual Conference of the International Speech Communication Association*, 2014. https://doi.org/10.21437/Interspeech.2014-80

[4] https://en.wikipedia.org/wiki/Dijkstra's algorithm.

[5] http://web.mit.edw/eranki/www/tutorials/search

[6] C. E. Rasmussen, *Processes for Machine Learning*. The MIT Press. 2006. https://doi.org/10.7551/mitpress/3206.001.0001

[7] L. R. Medsker, and L. C. Jain, *Recurrent neural networks. Design and Ap-plications*, 2001, 5.

[8] S. Bonnin, T. H. Weisswange, F. Kummert, and J. Schmuedderich, “General behavior prediction by a combination of scenario-specific models,” *IEEE Transactions on Intelligent Transportation Systems*, 15(4), 2014, pp. 1478–1488. https://doi.org/10.1109/TITS.2014.2299340.

[9] C. W. Hsu, C. C. Chang, and C. J. Lin, *A Practical Guide to Support Vector Classification*. Department of Computer Science, National Taiwan University, 2003.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *In Advances in Neural Information Processing Systems*. 2012, pp. 1097–1105).

Received October 11, 2022

Sineglazov Victor. ORCID 0000-0002-3297-9060. Doctor of Engineering Science. Professor. Head of the Department. Aviation Computer-Integrated Complexes Department, Faculty of Air Navigation, Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine.
Education: Kyiv Polytechnic Institute, Kyiv, Ukraine, (1973).

Research area: Air Navigation, Air Traffic Control, Identification of Complex Systems, Wind/Solar power plant, artificial intelligence.
Publications: more than 700 papers.
E-mail: svm@nau.edu.ua

Savchuk Serhii. Bachelor.
Aviation Computer-Integrated Complexes Department, Faculty of Air Navigation, Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine.
Education: National Aviation University, Kyiv, (2021).
Research area: air navigation.
Publications: 1.
E-mail: patcher_phoenix_0k@icloud.com

В. М. Синєглазов, С. Ю. Савчук. Підсистема формування польотного завдання

Дану роботу присвячено розробці системи автоматичного визначення польотного завдання для безпілотних літальних апаратів. Для реалізації такої системи використовувався фреймворк QT (бібліотека класів C++ та набір інструментального програмного забезпечення для створення крос-платформних додатків). До складу системи автоматизованого визначення польотного завдання запропоновано ввести модулі прогнозування та маршрутизації. Розроблено інтерфейс для побудови польотного завдання, яке може бути використане для завантаження, відстеження ключових метрик із безпілотного літального апарата та коригування місії польотного завдання. Було використано передові фреймворки QT/Qml, що дозволить використовувати програмний продукт на різних операційних системах, що додасть гнучкості у використанні системних компонентів.

Ключові слова: безпілотний літальний апарат; метод Калмана; алгоритм Дейкстри; інтерфейс користувача; обробка супутникових знімків.

Синєглазов Віктор Михайлович. ORCID 0000-0002-3297-9060.
Доктор технічних наук. Професор. Завідувач кафедрою.
Кафедра авіаційних комп’ютерно-інтегрованих комплексів, Факультет аеронавігації, електроніки і телекомунікацій, Національний авіаційний університет, Київ, Україна.
Освіта: Київський політехнічний інститут, Київ, Україна, (1973).
Напрям наукової діяльності: аеронавігація, управління повітряним рухом, ідентифікація складних систем, вітроенергетичні установки, штучний інтелект.
Кількість публікацій: більше 700 наукових робіт.
E-mail: svm@nau.edu.ua

Савчук Сергій Юрійович. Бакалавр.
Кафедра авіаційних комп’ютерно-інтегрованих комплексів, Факультет аеронавігації, електроніки та телекомунікацій, Національний авіаційний університет, Київ, Україна.
Освіта: Національний авіаційний університет, Київ, Україна, (2021).
Напрям наукової діяльності: аеронавігація.
Кількість публікацій: 1.
E-mail: patcher_phoenix_0k@icloud.com