University POLITEHNICA of Bucharest

Automatic Control and Computers Faculty,
Computer Science and Engineering Department

BACHELOR THESIS

Unmanned Aerial Vehicles, Formation Flight

Scientific Adviser:
Prof. Adina Magda Florea
Ing. Mihai Trăscău
Ș.l. Cătălin Leordeanu

Author:
Alexandru George Burghelea

Bucharest, 2013

Va multumesc tuturor pentru ajutorul acordat in elaborare

TODO:
SCRIE UN TODO

Abstract

Here goes the abstract about UAV Formation Flight.

TODO:
Write abstract

Contents

Notations and Abbreviations

ACS – Faculty of Automatic Control and Computer Science
ORCA – Optical Recognition Collision Avoidance
RC – Remote Control
SO – Self Organized
TNI – Teamnet International
UAS – Unmanned Aerial System
UAV – Unmanned Aerial Vehicle

Chapter 1

Introduction

1.1 Domain Description

In the last 3 decades the aeronautic industry has focused on creating methods of flying that does not involve a human factor inside the airplanes, developing solutions for unmanned flight. The necessity for advancements in domain is powered by the desire to keep human pilots out of harms way. UAV systems are useful in military missions, and high risk search and rescue missions. Along the military missions, UAVs can be used in civil context for missions like: traffic surveillance, cartography or animal tracking. An or drone is a vehicle that doesn't have a human pilot on board and can be either controlled by a , a ground control system (Control Tower) or be fully autonomous. The concepts of unmanned vehicles emerged a couple of years after the first mechanized flight in 1903 by Orville and Wilbur Wright [10]. In 1915 Nikola Tesla had a vision about a fleet of unmanned military aircrafts and in 1919 was developed the first UAV by Elmer Sperry, that was used for sinking a captured German battle ship. The first two countries that saw the high potential of unmanned vehicles were U.S.A and Israel. In 1960 the U.S. Air Force started a research program for developing UAVs, and in 1964 is the first documented use of an UAV in a real war scenario, during the Vietnam War. Israel started using UAVs for reconnaissance and surveillance mission. As a result, Israel reported no downed pilot during the Lebanon War in 1982. In the present, drones are intensively used in the war theaters from Afghanistan and Iraq [5]. The development of the autopilot is strongly correlated to the with the developed of the UAV. The company of Elmer Sperry was the first to produce an autopilot that was able able to fly autonomous for three hours in a straight line without being supervised by a human. By 1933 Sperry's autopilot was able to flight on true heading and maintaining the altitude, compared to the gyroscopic heading of the first version.The current evolution of autopilots is in close relation with the development of reliable communication systems. Although the first UAVs were controlled remotely by a human operator, they are now able to receive a flight plan and based on that to calculate the flight path and follow it to complete the mission. In moder autopilots the human factor the secondary role of supervising the system and controlling the on board equipment, like cameras and sensors.

1.1.1 Motivation

When I was a child I received my first toy airplane and became fascinated by the idea of moving freely like a bird.A couple of years later I first stood near a MIG-21 Lancer at my fathers garrison. The passion with witch the pilots talked about being in the air close to the clouds inspired me the love for moving freely in 3 dimensions.

The high number of human casualties reported in war theaters and training missions determined me to explore the field of autonomous fling. The necessity for reducing the loss of human lives gives autonomous great potential for evolution.

My motivation is to help create a next generation of autopilots capable of accomplishing difficult missions where it is the risk for a human pilot would not be affordable.

In Romania the UAV fields is still unexplored. The main fields where a UAV platform would be useful are interest points detection and monitoring, border patrol, search and rescue team and imagery intelligence.

1.1.2 Objectives

Although a single UAV is already able do accomplishing various mission by it's own, an interesting, and in my opinion mandatory, field is the one of fling in formation. A mission where the objective is to track multiple targets becomes very hard for a single drone, thus emerging the necessity for a swarm of UAVs. There are situation where the risk for loosing an UAV due to hostile conditions is to high, being more affordable to deploy multiple, cheaper UAVs in contrast to an expensive drone. Another use case for a formation would be a search mission where we can't equip a single aircraft with all the sensors necessary for success and choosing to use multiple specialized drones.

Usually a human pilot is able to fly in formation using a combination of cognitive and reactive behavior, always making small adjustments to maintain a coherent formation.

There are two ways that formation flight could be achieved. One would be a centralized method, where all the drones report the telemetry data to a central authority like a ground control system and the later would make the necessary decisions for all the involved actors and then relay the data back to the aircrafts. Although in theory this approach could give an optimal flight path, problems like delay in communication and sensors reporting faulty data could jeopardize the success of the mission. Another approach would be a decentralized method, inspired by swarms of animals, like ants or bees. In the second approach each UAV would decide what actions to execute based on the actions of the others.

The main goal of this thesis is to design an decentralized algorithm responsible for maintaining a flight formation based on the leaders actions. The leader will not share the flight path or mission plan with the other drones, it will share only the current position, speed and direction. Based only on this date, the drones must be able to maintain a predefined flight formation. Thus each drone, except the leader, is modeled as an reactive agent that has the mission to approach the leader and mimic his actions.

The secondary goal of this paper is to design a management platform for a fleet of airplanes that are able to execute different missions. The platform has the role of programming the mission for each drone, manage the in flight performance for each UAV and if necessary to send commands, inserted by a human supervisor, to the drones and by this modifying the current state of the mission execution.

The platform developed is possible thanks to a collaboration between the company and , University Politehnica Bucharest.

This thesis present my approach for UAV Formation Flight.

Chapter 2

Related Work

TODO:
Describe related work

A large number of articles describe the work done to solve the problem of autopilots UAV swarms, communication and coordinate systems exists. In the following paragraphs I will describe the main ideas and trade-offs for each described solutions.

The autopilot problem is well covered in the literature and from real life practice it is considered that the architecture is one that is stratified. Each layer is responsible for receiving commands from the superior layer, deciding if the the command would put the UAV in a state of imbalance or danger in general and forwarding the command to the layer below. If a possible imbalance state is detected the layer either discards the received command and does not send any command to the next layer, or it tweaks the command so that a incoherent state would not be induced.

The architecture as proposed by Borges de Sousa et. al [4] would have the following layers:

Platform
: The UAV vehicle with all the hardware

Maneuverer controller
: Controller that decides what hardware action is executed (roll, pitch, yaw)

Vehicle supervisor
: Basic autopilot capable to make simple decisions (setting the target speed, setting the heading)

Mission supervisor
: Artificial Intelligence System that plans the mission based on a template and the rest of the drones. It forwards the commands to the Vehicle Supervisor for validation

External controller
: Human factor that can interfere between the Mission Supervisor and Vehicle Supervisor to override or even deactivate the first one.

The architecture imagined by Borges de Souse can be seen in Figure 2.1.

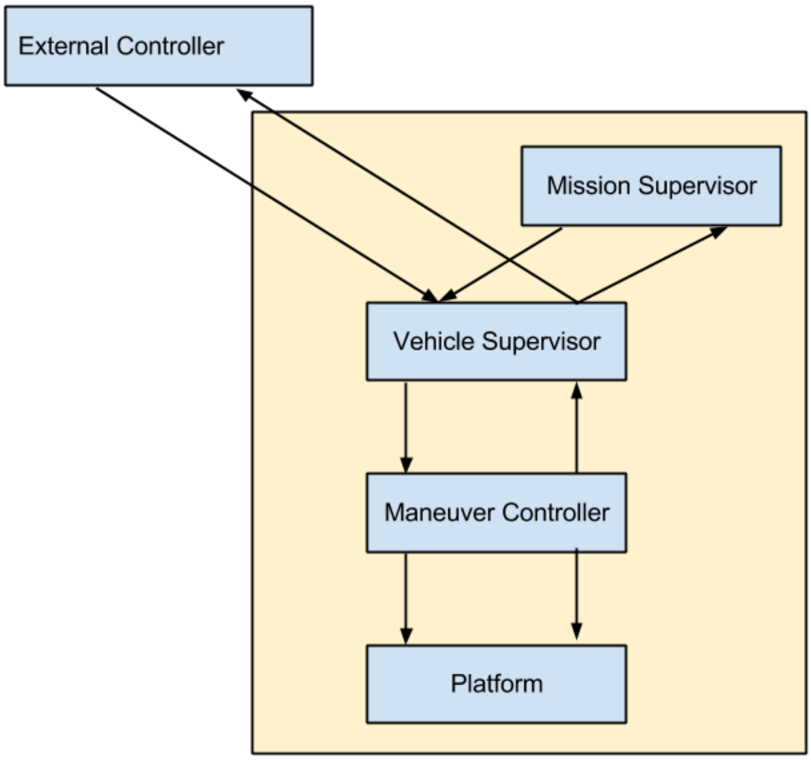

Figure 2.1: Vehicle Control Architecture

The advantages provided by this kind of architecture is the separation of concerns for each level and the fact that the repair in case of failure can be easily detected and fixed. The only draw-back of this approach is the fact that in the case of poor implementation it could introduce a latency in communication loosing the possibility of having a real time system.

In the implementation my autopilot system is similar to the one described above because the Hirrus drone [11] provided by TNI already contains a maneuver controller,vehicle supervisor and an external controller. Thus my system would act as a mission supervisor.

In the terms of UAV teams, Mark D. Richards and his colleagues [9] identify two main groups of strategies. The first one, called **Behavior-based Control systems**, use a mesh of interacting high-level behaviors to perform a task. The second one, **Deliberative System**, acts by creating a specific flight path for each individual UAV to follow. This second behavior is a generalization of the search path optimization that Ablavsky proposes in [1]. Ablavsky approaches the problem by following the following steps:

1. Restrict the search area based on the mobility of the target.
2. Divide the search area in to the smallest number of sub-regions keeping in mind the constrains of the aircraft.
3. Determine a search pattern for each sub-region with the property of assuring full coverage with a minimized path length.
4. Combine the individual results into an optimal global path.

The deliberative approach used by Richards is based on dividing an area for each UAV and generating inside the designated zone a flight path to be followed. For achieving some degree of flexibility, Richards opted not to use an adaptive replanning where a central controller computes a specific flight path for each agent and and then broadcasts it to the team. The drawbacks of this approach are represented by the fact that there would still be a single point of failure and the fact that by the time the plan is sent to the UAV it may be already be deprecated. The approach that was used was a reactive one. The initial path is computed and if an hostile condition is generated each UAV has to determine a way to exit that state. By these means Richards managed to sweep an area that is divided in sub-zones with different degrees of danger and even a no fly zone. The reactive behavior was useful for avoid collision with a friendly unit or for escaping a danger zone.

Although Richards proposes uses a long range unsynchronized flight team, in my implementation I used a similar approach for obtaining a synchronized formation.

Gaudiano and his team research the possibility to explore a zone based on the strategies used by schools of fish, flocks of birds and swarms of insects. The research they published in [?] and [?] is based on mimicking the pheromones left behind by the swarms of insects. To be able to replicate the insects behavior, Gaudiano assumed that each UAV is aware of the terrain geometry, is equipped with a sensor that provides a forward code of vision able to detect elevation and distance, has a sensor to detect the other UAVs in a given sphere, is positioned using GPS-like coordinates and has a sensor that mimics the pheromone detection. The later sensor detects, within a rectangular region centered on the UAV, the coverage of the current cell. In their simulation they have used a global communication system that was able to assure data flow between each two UAVs. The strategy to control the UAVs is implemented in a decentralized way where each UAV can decide what action to execute. Gaudiano tested 5 strategies to explore an area:

Baseline
: Each UAV starts with a different heading, flies in a straight line until the area limit is reached and then turns so that it doesn't leave the area.

Random
: Is similar to the baseline implementation but at each time step, the UAV will change it's heading by a small random angle.

Repulsion
: An UAV can detect the other UAVs on a given radius and it changes it's heading so that they do not intersect.

Pheromone
: Each UAV is able to detect the already explored cells and changes it's heading so that it heads in the direction of unexplored cells.

Global
: The area is divided in smaller regions an an UAV know how many UAVs are in ore region and based on that it decides what region to explore.

Based on their experiments they have discovered that the most efficient way to obtain a higher coverage is the pheromone strategy. Compared to the repulsion strategy that obtained a 44% coverage while the pheromone strategy obtained a 60%. These coverages were obtained using 10 UAVs. The same experiments showed that scaling to larger UAV swarms presents a drawback represented by the fact that although the coverage efficiency increases with the number of UAVs, their relative efficiency decreases. While on a very large area 10 UAVs would cover 6%, 110 UAVs would cover only 33%.

In contrast with the approaches form above, Nowak centered his research on creating a self organized swarm. To achieve this he created a two tired architecture for a System Model that separates the implementation details from the real world agents behavior. This architecture (as presented in [7]) can be seen in Figure 2.2.

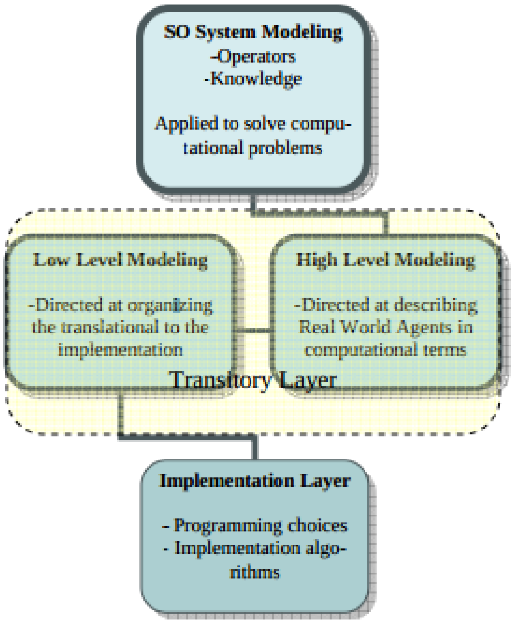

Figure 2.2: Three tier SO architecture

The behavior of many cooperative agents in the same are is a behavior that has been studied in detail by many researchers. Reynolds proposed in 1987, three fundamental rules for swarming: [8]

Separation
The UAV must steer away from the nearest neighbors to avoid crowding.

Cohesion
The UAV must steer towards the average position of its neighbors.

Alignment
The UAV must steer towards an average heading of the neighbors.

A visual representation can be seen in Figure 2.3

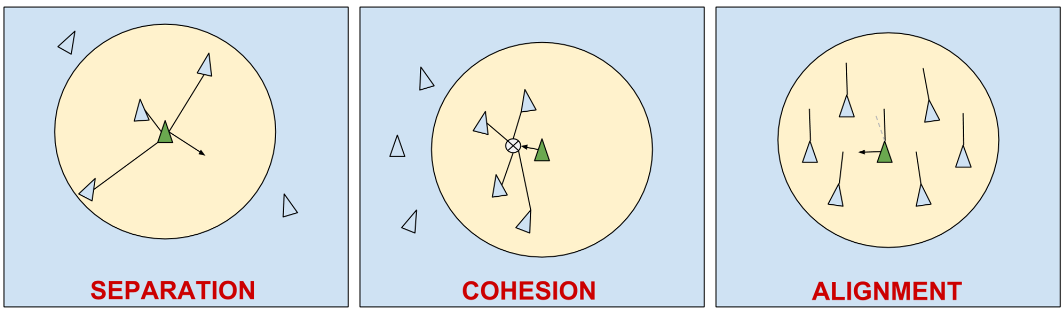

Figure 2.3: Fundamental Swarming Rules

Based on Reynolds rules, Nowak implemented in SWARMFARE 10 rules for a coherent formation flight [7]:

Flat Align
vector align neighbors

Target Orbit
orbit target at safe distance

Cluster Range Towards
cohesion

Cluster Range Away
separation

Attract
towards center of mass of all targets

Weighted Attract
towards closest target

Target Repel
repel if with 90% of UAV sensor range

Weighted Target Repel
repulsion based on proximity to target

Evade
Apriori collision detection and avoidance

Obstacle Avoidance
Real time obstacle avoidance

The difference between the previous approaches and Nowaks is that the later considers at no point an agent will give commands such as "Slow down", "Turn" or "Dive". Each UAV is responsible for computing what action to execute based on a minimal set of date received from the other UAVs. A possible set of data is formed from:

- GPS coordinates;
- altitude;
- speed; and
- heading

To test the SWARMFARE framework, Nowak opted to paralyze the computation using a Master-Slave architecture. The resulting parallel time calculation results from the equation:

$$T_p = k(t_s + t_w m)(p-1) + \frac{n^{ck}}{p} \qquad \begin{aligned} k &= \text{number of generations} \\ t_s &= \text{setup time} \\ t_w &= \text{transfer time} \\ p &= \text{Number of processors} \\ n &= \text{Number in population} \\ p &= \text{constant defining the variable} \\ n^c k &= \text{linear calculation time} \end{aligned} \tag{2.1}$$

The time obtained by using a initial population of 2 per node is 540 minutes for one node and 80 minutes for 16 nodes.

The SWARMFARE is a work in progress and the trade-offs that could appear only from the implementation of the 10 rules.

The Autonomous UAV project proposes to implement an autopilot that integrates the behaviors depicted by the rules proposed by Nowak. The formation flight module implement in this thesis will integrate with the obstacle avoidance module and surveillance modules developed by my colleagues.

Based on the communication mode, there are two main methods for implementing UAV formation flight: coordinated or cooperative. Bourgault et. al. depict the difference between the two modes in [2]. According to them in coordinated approach each agent decides what actions to execute based on the current knowledge of the world but there is no possibility that the actions of an agent to influence the decisions of another. The cooperative mode has the advantage that agents negotiate what actions to execute, thus obtaining a global optimal solution. On the other side the coordinated mode obtains a suboptimal solution, but has the advantage that the individual computation time does not increase with the number of agents.

For a close range formation there are necessary both these modes. The coordinated mode is useful for computing the position in the formation for each aircraft and the cooperative mode is useful for a collision avoidance module so that the aircrafts do not collide. Similar to Bourgault I implemented the coordinated mode with a one-step look-ahead.

For obtaining obstacle avoidance it's necessary to have a reactive system combined with a co-operative decision making algorithm. Call et. al tried to implement such a system by obtaining data from video cameras. The cameras are preferred because they are passive, lightweight, simple and inexpensive sensors. The drawback for these sensors is represented by the fact that under low-visibility, such as fog, smoke or night, have reduced efficiency. Also the processing of the information could have high computation time. In the research paper [3], they have managed to obtain three dimensional data from cameras either by using a stereo vision system or by using multiple frames from one camera. They have found out that the relative position of an object can be obtained with a an enough amount of information by using a single camera and multiple frames. The main problem with detecting obstacles is that most of the existing algorithms are based on detecting a texture variation or the detection of corners. Problems in detecting obstacles appear when the objects are described by parallel lines with no visible endpoint or have no texture. Another case when the algorithm could fail is when the obstacles have reflective glass surfaces.

The results published in [3] can be seen Figure 2.4

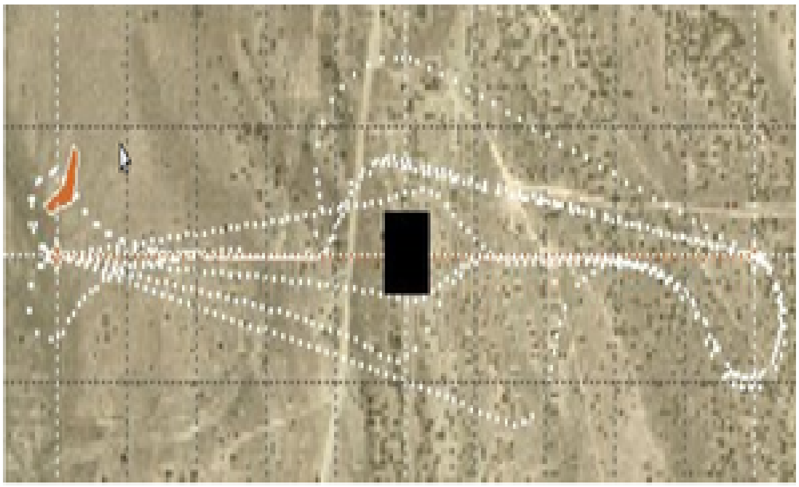

Figure 2.4: Overhead view of the flight path of the UAV as it avoids an obstacle indicated by the black box

When using cameras to detect obstacles, the angular velocity of gyroscope often has a large amount of noise, mostly because of the yaw angle or wind. Thus the angular velocity has to be estimated using another type of on-board sensor.

He approaches this problem by following the following steps [6]:

1. Classify the image blocks in structural and non-structural maps
2. Find multiple "best" motion vectors instead of a single one and chose the best one using a predefined metric
3. Using the data about structural motion, estimate the camera motion to reduce uncertainty in the motion for non-structural blocks.

In the same article [6] uses the following two formulas to compensate the motion noise of the camera:

$$m(O_B, r) = \frac{1}{|C(O_B, r|} \oint_{C(O_B,r)} I_t(x, y)dxdy, 0 < r \leq Rs \tag{2.2}$$

Where:

- $O_B = (X_B, Y_B)$ center of block B
- $C(O_B, r)$ = circle center in O_B with radius r
- R = maximum radius to search
- $m(O_B, r)$ = intensity profile of pixel O_B or block B

$$d_1(A, B) = \min_{1-\delta\leq\lambda\leq 1+\delta} \max_{0\leq r\leq \frac{R}{\lambda}} |m(O_A, \lambda r) - m(O_B, r)| \tag{2.3}$$

Where:

- λ = scaling factor
- $[1 - \delta, 1 + \delta]$ = search range for λ

From Equiation 2.2 and Equiation 2.3 the distance between blocks A and B can be computed with the following formula:

$$d(A, B) = wd_0(A, B) + (1 - w)d_1(A, B) \qquad w = \text{ weighted factor} \tag{2.4}$$

The formation flight module implemented for this thesis will integrate with a module of collision avoidance based on the

Chapter 3

UAV Management Platform

3.1 Architecture

TODO:
Describe architecture

3.2 Functionalities

TODO:
describe functionalities

TODO:
Describe management framework

Chapter 4

Formation Flight

4.1 Coordinates Systems

4.1.1 Latitude, Longitude, Altitude

TODO:
describe lla

4.1.2 Earth-Centered, Earth-Fixed

TODO:
describe Ecef

4.1.3 Conversion

TODO:
describe lla-to-ecef

4.2 Formation types

TODO:
describe formation types

4.3 Entering the formation

TODO:
Describe Formation Entering

4.4 Maintaining the formation

TODO:
Describe entering the formation

Chapter 5

Implementation details

TODO:
describe implementation details

5.1 FlightGear

TODO:
talk about FlightGear

5.2 QGroundControl

TODO:
talk about QGroundControl

5.3 MY CODE

TODO:
talk about code, class-diagrams, describe methods, data-structures

Chapter 6

Use cases

TODO:
describe use cases

TODO:
captures of 3 formations

TODO:
captures of flight-path

TODO:
possible missions

Chapter 7

Conclusions and future work

TODO:
describe conclusions and future work with low priority

TODO:
Add appendixes

Bibliography

[1] V. Ablavsky, D. Stouch, and M. Snorrason. Search path optimization for uavs using stochastic sampling with abstract pattern descriptors. AIAA Guidance Navigation and Control Conference, August 2003.

[2] Frédéric Bourgault, Tomonari Furukawa, and Hugh F. Durrant-Whyte. Coordinated decentralized search for a lost target in a bayesian world. Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems, October 2003.

[3] Brandon Call, Randy Beard, Clark Taylor, and Blake Barber. Obstacle avoidance for unmanned air vehicles using image feature tracking. American Institute of Aeronautics and Astronautics, 2006.

[4] J. Borges de Sousa, G. Goncalves, A. Costa, and J. Morgado. Mixed initiative control of unmanned air vehicle systems: the pitvant rd uav programe. Portuguese Ministry of Defense under the project PITVANT, 2008.

[5] M. E. Dempsey. Eyes of the army - u.s. army roadmap for unmanned aircraft systems 2010-2035. U.S Army, Fort Rucker, 2010.

[6] Zhihai He. Vision-based uav flight control and obstacle avoidance. Proceedings of the 2006 AIAA Guidance, Navigation, and Control Conference, August 2006.

[7] Dustin J. Nowak, Ian Price, and Gary B. Lamont. Self organized uav swarms planning optimization for search and destroy using swarmfare simulation. Proceedings of the 2007 Winter Simulation Conference, 2007.

[8] C. Reynolds. Boids - background and update. http://www.red3d.com/cwr/boids/, September 2001.

[9] Marc D. Richards, Darell Whitley, and J. Ross Beveridge. Evolving cooperative strategies for uav teams. Department of Computer Science, Colorado State University, June 2005.

[10] R. Storm and T. Benson. Learning to fly: The wright brothers' adventure. National Aeronautics and Space Administration, 2002.

[11] Another Flight Team. Hirrus, 2012.