

Politechnika Opolska  
Wydział Elektrotechniki, Automatyki i Informatyki  
Katedra Elektrotechniki i Mechatroniki



# **POLITECHNIKA** OPOLSKA

## **AUTOREFERAT PRACY DOKTORSKIEJ**

Dyscyplina: *Automatyka, Elektronika, Elektrotechnika i Technologie Kosmiczne*

Temat:

**STEROWANIE ZDECENTRALIZOWANĄ SIECIĄ  
BEZZAŁOGOWYCH JEDNOSTEK LATAJĄCYCH**

Promotor:  
dr hab. inż. Andrzej Waindok, prof. Uczelni

Pracę wykonał:  
mgr inż. Patryk Szywalski

Opole 2022

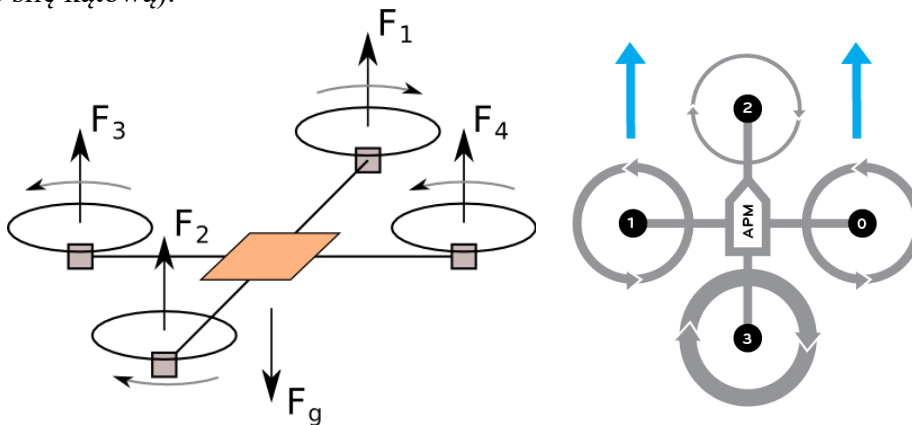
# Spis treści

<b>1. Wstęp .....</b>	<b>2</b>
<b>2. Cel i zakres pracy .....</b>	<b>3</b>
<b>3. Bezzałogowy system latający (UAS) .....</b>	<b>4</b>
3.1 <i>Konstrukcja mechaniczna UAV.....</i>	5
3.2 <i>Struktura bezzałogowego systemu latającego.....</i>	7
3.3 <i>Systemy pozycjonowania wykorzystane w urządzeniu .....</i>	8
3.4 <i>Określenie zwrotu w układzie współrzędnych.....</i>	9
<b>4. Decentralizacja sieci radiowej.....</b>	<b>10</b>
4.1 <i>Założenia strukturalne sieci radiowej .....</i>	10
4.2 <i>Zasada działania sieci radiowej.....</i>	10
4.3 <i>Testy sieci radiowej .....</i>	11
<b>5. Sterowanie pojedynczym dronem (UAV) oraz grupą dronów (UAS) .....</b>	<b>12</b>
5.1 <i>Wybór sposobu sterowania UAV.....</i>	12
5.2 <i>Układy współrzędnych.....</i>	12
5.3 <i>Generowanie trajektorii.....</i>	13
5.4 <i>Algorytm sterowania .....</i>	15
<b>6. Testy pojedynczego UAV .....</b>	<b>17</b>
6.1 <i>Wzniesienie .....</i>	17
6.2 <i>Lot po punktach przelotowych .....</i>	17
<b>7. Laboratorium Autonomicznych Systemów Latających.....</b>	<b>19</b>
7.1 <i>Konstrukcja sali.....</i>	19
7.2 <i>System nawigacji .....</i>	20
7.3 <i>Wyznaczenie zwrotu urządzenia.....</i>	20
7.4 <i>Hardware – panel obsługi .....</i>	21
7.5 <i>Software – interfejs operatora .....</i>	22
<b>8. Algorytm dla przestrzeni otwartej.....</b>	<b>23</b>
8.1 <i>Algorytm: Dobór w pary.....</i>	23
8.2 <i>Symulacja lotu.....</i>	25
<b>9. Algorytm dla przestrzeni zamkniętej .....</b>	<b>26</b>
9.1 <i>Algorytm: rozwiązanie labiryntu metodą analityczną.....</i>	26
9.2 <i>Symulacja lotu.....</i>	28
<b>10. Badania na obiektach rzeczywistych .....</b>	<b>29</b>
10.1 <i>Sterowanie zależne .....</i>	31
10.2 <i>Sterowanie niezależne.....</i>	35
<b>11. Wnioski i uwagi .....</b>	<b>39</b>
<b>12. Wybrane pozycje ze spisu literatury rozprawy doktorskiej .....</b>	<b>40</b>

## 1. Wstęp

Bezzałogowe Urządzenie Latające (ang. Unmanned Aerial Vehicle – UAV) do wykonania ruchu (zadania) nie wymaga ingerencji człowieka. Urządzenia te potocznie nazywane są dronami. Operator, sterujący ruchem urządzenia, najczęściej komunikuje się z nim drogą radiową przy wykorzystaniu do tego celu dedykowanej aparatury. Obecnie możliwe są już systemy, które działają automatycznie bądź autonomicznie. W takim wypadku operator może nadzorować pracę urządzenia, jednak ruchem zarządza jednostka sterująca znajdująca się na urządzeniu [X2]. W literaturze potocznej często sformułowanie automatyczne/autonomiczne błędnie pada zamiennie. Jako lot automatyczny rozumie się ruch po wcześniej zaplanowanej trajektorii. Urządzenie nie ma prawa zmienić tej trajektorii. Ruch autonomiczny daje urządzeniu pełną swobodę podczas ruchu, pozwalając mu na dobrowolny wybór trajektorii, zakładając jedynie punkt docelowy, względnie parametry graniczne np. prędkość przelotową bądź możliwą przestrzeń ruchu. Drugim, często używanym pojęciem jest Bezzałogowy System Latający (ang. Unmanned Aerial System, UAS). Jako UAS określano zbiór kilku współpracujących UAV połączonych w jeden system. W skład UAS wchodzi urządzenia latające UAV, ale także cała aparatura naziemna [X15]. Podział zadań w obrębie UAS o ograniczonych zasobach (UAV) w nieznanym środowisku jest nadal trudnym problemem, zważywszy na ograniczone informacje o bardzo dynamicznym środowisku, brak ciągłej i niezawodnej sieci komunikacyjnej oraz ograniczenia związane z dostępną energią [15].

W ramach pracy wykorzystywano czterosilnikowe urządzenia o pionowym starcie, potocznie nazywane quadcopterami. Taka konstrukcja pozwala na lot tylko i wyłącznie przy zastosowaniu odpowiednich algorytmów sterowania. Rozumie się przez to, że niemożliwe jest ręczne sterowanie (prędkościami obrotowymi) czterech silników w celu uzyskania kontrolowanego lotu jak np. przy samolocie (sterując gazem, oraz lotkami) bądź helikopterze (sterując gazem, a mechanizm różnicowy skompensuje siłę kątową).



**Rys. 1.1.** Wpływ prędkości obrotowej oraz kierunku obrotu jednostek napędowych na ruch UAV  
*a) wznoszenie i opadanie, b) ruch do przodu [29]*

Quadcopter poza pionowym startem i lądowaniem cechuje się także możliwością zwisu tzn. zatrzymaniu się w powietrzu oraz bezpośredniego ruchu w każdą ze stron. Nie jest wymagany do tego (jak np. w przypadku samolotu) obrót maszyny. Aby urządzenie się podniosło (rys. 1.1a) siła generowana przez cztery jednostki napędowe  $F_1 - F_4$  musi być większa od siły grawitacyjnej  $F_g$  działającej na urządzenie. W celu kompensacji powstałej siły kątowej przeciwległe silniki kręcą się parami w jedną, a kolejne dwa w drugą stronę. Aby kierunek generowanej siły był zgodny z założonym, stosuje się dwa rodzaje śmigieł: CW (Clock Wise) oraz CCW (Counter – Clock Wise). Lot poziomy w stosunku do osi Ziemi (rys. 1.1b) otrzymuje się poprzez doprowadzenie urządzenia

do przechylenia o pewien kąt. Efekt ten uzyskuje się poprzez zwiększenie generowanej siły tylnych silników w stosunku do przednich.

Wykorzystywane urządzenie pozwala na obroty wokół swojej osi:

- $x$  – przechylenie (ang. bank lub roll) to ruch UAV wokół osi podłużnej,
- $y$  – pochylenie (ang. pitch) to ruch UAV wokół osi poprzecznej,
- $z$  – odchylenie (ang. yaw) to ruch UAV wokół osi pionowej.

Istnieje wiele metod pozwalających na wyznaczenie pozycji urządzenia w przestrzeni R3. Najpopularniejszymi systemami są:

- Globalne systemy nawigacji satelitarnej – GNSS (z ang. Global Navigation Satellite Systems) [11, 23, 27, 28].
- Nawigacja inercyjna [5, 12, 13, 22].
- Lokalny system pozycjonowania – LPS (z ang. Local Positioning System) [14, 26].
- Pomiar dźwięku [2, 8].
- Pomiar fal radiowych (WiFi, Bluetooth, ZigBee, UWB) [1, 3, 4, 19, 20].
- Graficznie na podstawie analizy obrazu np. *magiczny dywan* [16, 14].

## 2. Cel i zakres pracy

W ramach pracy podjęto się wykonania bezzałogowego systemu latającego, składającego się z pięciu urządzeń. Poprzez stwierdzenie *bezzałogowy system latający (UAS)* rozumie się grupę bezzałogowych urządzeń latających (UAV) potrafiących kooperować w celu wykonania powierzonego im zadania. Ważnym elementem jest praca systemu w sposób zdecentralizowany. Rozumie się przez to pracę systemu nieskoncentrowanego na jednym urządzeniu (np. przy transmisji danych czy wyznaczeniu ścieżki lotu), a na całej sieci, traktując każdą jednostkę jako równouprawnionego członka. Zaletą takiego rozwiązania, w porównaniu do systemów zdecentralizowanych, jest to, że w razie awarii dowolnego UAV, bądź chwilowej utraty łączności, system pracuje nieprzerwanie.

Zakres pracy podzielono na trzy główne obszary:

I. Konstrukcja bezzałogowej jednostki latającej, która obejmowała:

- Projekt i wykonanie dwóch ram urządzenia (klasy 130 i 300). Na podstawie symulacji określono siły działające na ramę oraz dobrano odpowiednie materiały w celu zapewnienia wymaganej wytrzymałości i sztywności. Celem pośrednim przy projekcie ramy było wykorzystanie druku 3D w technologii FDM w celu minimalizacji masy i ceny. W ramach pracy wykorzystano głównie takie materiały jak: ABS, PLA, TPU, NanoCarbon (PA 12 + CF) i PETG.
- Projekt i wykonanie części elektrycznej UAV. Obejmował on dobór jednostki sterującej, komunikacji, zasilania, jednostki napędowej i systemu pozycjonowania.
- Zbadanie algorytmów sterowania i stabilizacji pojedynczej jednostki latającej.
- Badania związane z określeniem zwrotu urządzenia wewnątrz budynku.
- Badania systemów nawigacji: systemów satelitarnych, batometrycznego czujnika wysokości, LIDARu, nawigacji lokalnej opartej na stacjonarnych sygnałach ultradźwiękowych połączonych z interfejsem radiowym 433 MHz.
- Napisanie programu do planowania i generowania trajektorii do lotów automatycznych.

II. Budowa Laboratorium Autonomicznych Systemów Latających (LASL), która obejmowała:

- Przygotowanie pomieszczenia do badania lotów dronów.

- Projekt i wykonanie sali laboratoryjnej na wzór Szwajcarskiego Flying Machine Arena. Wykonano część mechaniczną (konstrukcja namiotowa + siatki, klatki bezpieczeństwa, słupy nawigacyjne), a także przeprowadzono montaż i uruchomienie systemu pozycjonowania.
- Wykonanie panelu sterowania wraz z odpowiednimi zabezpieczeniami w celu zapewnieniu wykonywania bezpiecznych lotów.
- Napisanie programu nadzorującego pracę bezzałogowego systemu latającego.
- Wykonanie sieci radiowej do komunikacji wewnątrz UAS w sposób zdecentralizowany.

### III. Algorytmy sterowania grupą dronów:

- Sformułowanie algorytmu do wyznaczania optymalnej trajektorii dla UAV i UAS w przestrzeni otwartej i zamkniętej.
- Wykonanie algorytmu zmieniającego strukturę (pozycję) sieci w sposób skoordynowany.
- Sformułowanie algorytmu do znalezienia drogi wyjścia z labiryntu

Po realizacji wymienionych zadań możliwe było wykonanie i przetestowanie algorytmów z punktu trzeciego na pięciu fizycznych urządzeniach. Zgodnie z przyjętą w rozdziale pierwszym klasyfikacją poziomów autonomii, założono, że system powinien umożliwiać sklasyfikowanie go na 4 (z 5-ciu) poziomie.

Założono także, że powstałe laboratorium powinno pozwalać na szczegółowe badania bezzałogowego systemu latającego pod kątem szeroko pojętego sterowania, wytrzymałości konstrukcji oraz wpływu wykorzystanych komponentów na jakość lotu (np. baterie Li-pol, Li-ion i inne).

Uwzględniając powyższy zakres rozprawy sformułowano tezę pracy, która brzmi:

### **Możliwe jest zdecentralizowane sterowanie bezzałogowego systemu latającego z wykorzystaniem algorytmów deterministycznych.**

Realizacja założonych celów pracy wymagała udowodnienia tez pomocniczych:

1. Wykorzystanie sieci radiowej pozwala na komunikację w ramach zdecentralizowanej grupy bezzałogowych urządzeń latających.
2. Możliwe jest użycie deterministycznych algorytmów do wyznaczenia trajektorii oraz rozwiązania problemu labiryntu, tzn. znalezienia drogi wyjścia z labiryntu dla bezzałogowego systemu latającego.

## **3. Bezzałogowy system latający (UAS)**

W ramach pracy zaprojektowano, zbadano i wykonano dwie konstrukcje bezzałogowego urządzenia latającego. Pierwotnie założono, iż konstrukcja ma być możliwie mała, ale pozwalająca na montaż odbiornika GNSS i charakteryzująca się czasem lotu minimum 10 minut. Ze względu na stosowany system pozycjonowania (satelitarny), początkowo loty były możliwe tylko na zewnątrz. Podczas projektu korzystano z druku 3D w technologii FDM (z ang. Fused deposition modeling). Przegląd literaturowy wskazał na duży potencjał druku w lotnictwie. Poza wydrukiem gotowych modeli do badań w tunelu aerodynamicznym, możliwe jest także drukowanie poszczególnych elementów UAV, w tym kół bądź skrzydła (w samolotach) [9, 17, 21]. Badania wstępne zakładały testy wytrzymałościowe konstrukcji z wykorzystaniem materiałów PLA, ABS, PETG, TPU, NanoCarbon (PA 12 + CF) [X8].

Wraz z rozwojem projektu zdecydowano się przenieść badania do zamkniętego pomieszczenia. Wymagało to zmiany systemu pozycjonowania urządzenia, ze względu na brak połączenia odbiornika satelitarnego z wymaganą liczbą satelitów w budynku. Ostatecznie, głównie ze względu na dokładność pomiaru ( $\pm 2\text{cm}$ ) oraz cenę systemu zdecydowano się na rozwiązanie firmy Marvelmind.

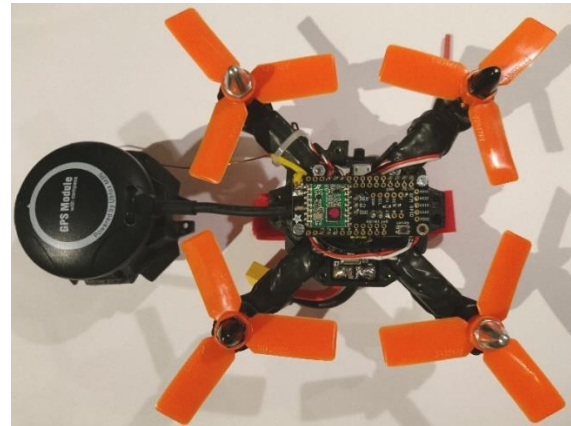
### 3.1 Konstrukcja mechaniczna UAV

#### 3.1.1. Urządzenie klasy 130

Konstrukcja klasy 130 była pierwszym urządzeniem wykonanym w celu realizacji postawionych w rozprawie założeń. Jej projekt pozwolił na przetestowanie nowych technik i metod produkcji urządzeń. Na rysunku 3.1 pokazano zbudowane urządzenie. Rysunek 3.2 pokazuje zabudowany sterownik znajdujący się w centralnej części urządzenia. Nogi konstrukcji zostały wydrukowane z materiału gumo-podobnego – TPU.

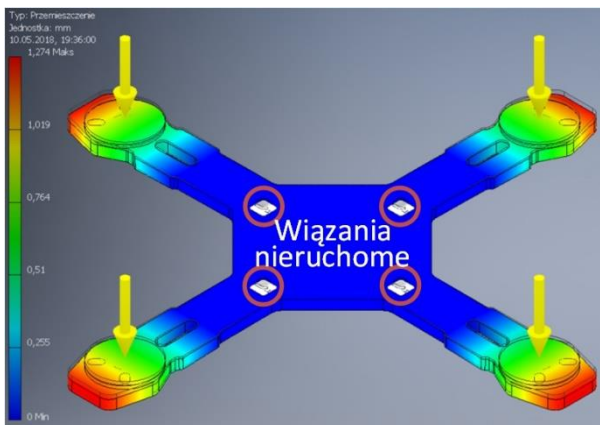


Rys. 3.1. Widok z boku

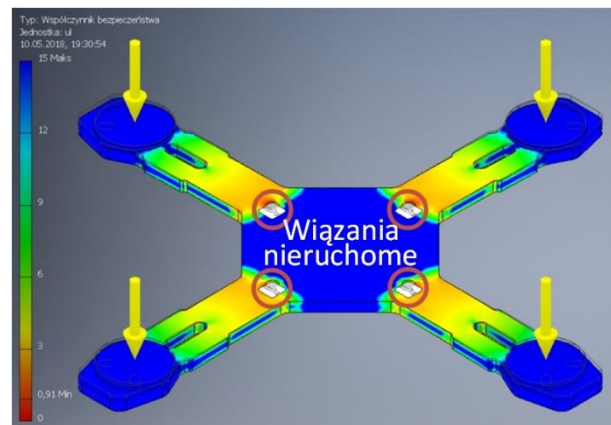


Rys. 3.2. Montaż elektroniki – widok z góry

Na rysunku 3.3 pokazano uzyskane w symulacji miejsca na płycie nośnej (kolor czerwony), w których na skutek siły, przemieszczenie jest największe. Pod wpływem działania siły, krańce ramion mogą wygiąć się o 1,24 [mm]. Rysunek 3.4 przedstawia to samo badanie jednak pod kątem naprężeń pojawiających się podczas nacisku.



Rys. 3.3. Symulacja obciążenia płyty nośnej wykonanej z GFRP siłą 14 N na ramię – przemieszczenie



Rys. 3.4. Symulacja obciążenia płyty nośnej wykonanej z GFRP siłą 14 N na ramię – współczynnik bezpieczeństwa

Podsumowując, badania wykazały, że jest możliwe wykorzystywanie druku 3D przy projektowaniu ram bezzałogowych urządzeń latających. Konstrukcja klasy 130 jest w pełni funkcjonalna i może zostać użyta w ramach dalszych badań. Jediną jej wadą jest, ze względu na możliwe niewielkie dociążenie urządzenia, niemożność stosowania lokalnego systemu pozycjonowania (LPS).



### 3.1.2. Urządzenie klasy 300

Wraz z rozwojem projektu, zdecydowano o przeniesieniu badań do pomieszczenia zamkniętego. Motywacją zbudowania dedykowanego laboratorium była między innymi ograniczona dokładność stosowanego systemu satelitarnego pozycjonowania, która – nawet dla najlepszego odbiornika – w stanie statycznym wynosiła 2 metry.

Na rysunku 3.5 widać zamontowany na karbonowym drążku magnetometr. Czujnik wyciągnięto celowo ponad osie silników minimalizując ich wpływ na wskazanie. Montaż sensora z dala od pozostałej elektroniki, szczególnie kabli wysokoprądowych, jest konieczny w celu uzyskania niezakłóconego pomiaru. W ramach badań zauważono także zaburzenie pomiaru bezpośrednio przy ziemi w laboratorium.



*Rys. 3.5. Bezzałogowe urządzenie latające o czwartym numerze identyfikacji ( $id = 4$ ) – widok od przodu*

Zgodnie z założeniami pracy powstało pięć bezzałogowych urządzeń latających klasy 300. Na rysunku 3.6 pokazano ich ostatnie wersje konstrukcyjne. Każde urządzenie ma indywidualny kolor oraz numer identyfikacyjny.



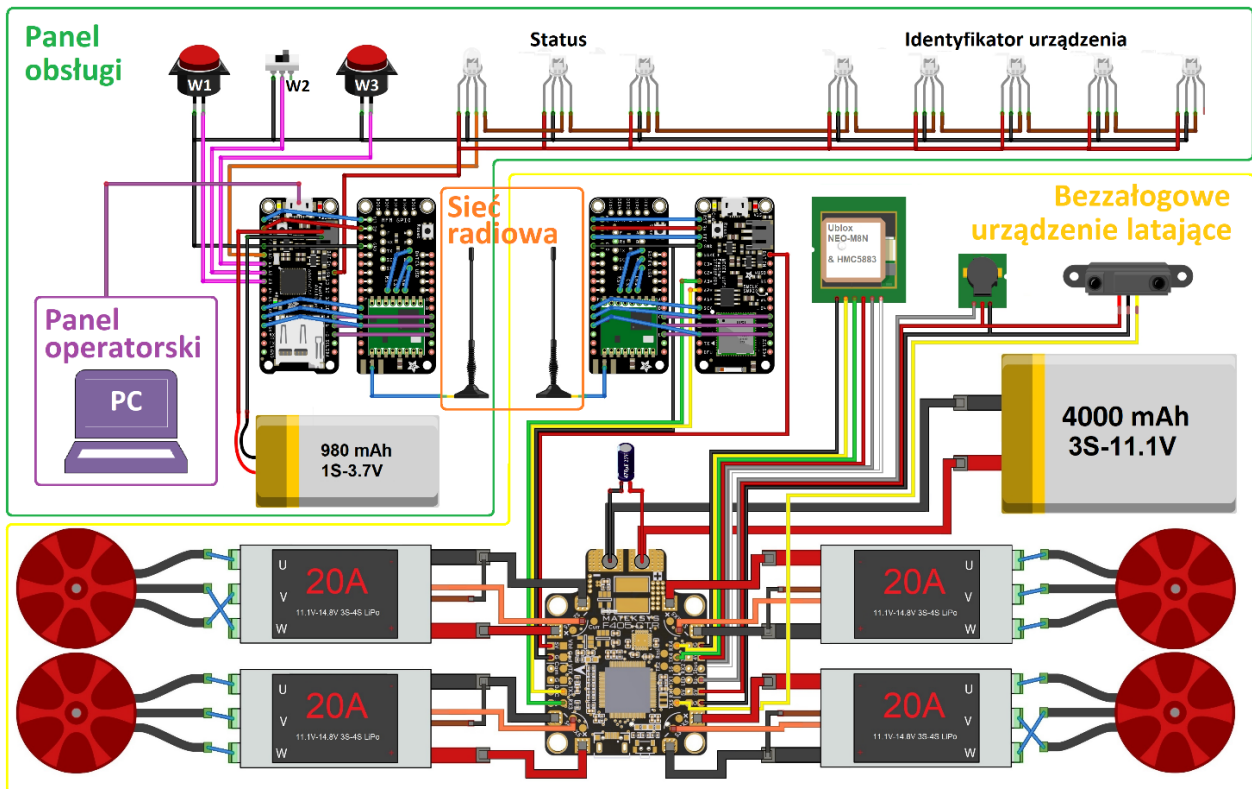
*Rys. 3.6. Wykorzystywane w badaniach bezzałogowe urządzenia latające*

### 3.2 Struktura bezzałogowego systemu latającego

Bezzałogowy system latający podzielono na cztery główne elementy (rys. 3.7, 3.8):

- Bezzałogowa jednostka latająca bądź ich zbiór – kolor żółty,
- System komunikacji radiowej – kolor pomarańczowy,
- Panel obsługi – kolor zielony,
- Panel operatorski, interfejs użytkownika na komputerze klasy PC – kolor fioletowy, (opcjonalnie)

Na rysunku 3.7 przedstawiono poglądowy schemat elektryczny bezzałogowego systemu latającego, dla przykładu z jednym bezzałogowym urządzeniem latającym. Aby zachować czytelność, kolory opisujące wybrane elementy zostały zachowane także w późniejszym opisie struktury pracy systemu oraz na schemacie blokowym (rysunek 3.8).



Rys. 3.7. Schemat połączeń układu

Bezzałogowa jednostka latająca zawiera w swojej budowie dwa procesory, czujniki oraz moduł komunikacji. Zadania podzielono pomiędzy dwoma procesorami, co daje szereg zalet:

- Procesor pierwszy pracuje w razie awarii drugiego niezależnie, tzn. UAV w razie awarii drugiego procesora jest nadal w pełni sterowalny.
- Możliwość stosowania gotowego oprogramowania typu open-source do stabilizacji urządzenia na procesorze pierwszym.
- Ociążenie pierwszego, jednorodzeniowego procesora.
- Modułowość systemu – każdy procesor ma swoje zadania zgodnie z schematem blokowym na rysunku 3.8. Modyfikacja kodu procesora drugiego nie powinna doprowadzić do nieplanowej awarii procesora nadrzędnego.

Zwazwszy na względy bezpieczeństwa, prezentowane rozwiązanie, pomimo wady w postaci dodatkowej masy jednego modułu oraz problemów komunikacyjnych pomiędzy procesorami,



okazało się lepsze niż rozwiązanie z jednym procesorem. Na rysunku 3.8 uwzględniono priorytety kontrolera lotu. Schemat na rys. 3.7 przedstawia połączenia przy wykorzystaniu odbiornika GNSS. Podczas badań wewnątrz laboratorium *LASL*, UAV korzystało z systemu *Marvelmind*. Zamontowany mobilny odbiornik sygnału był kompatybilny pod względem przyłącza oraz magistrali danych.

Id w sieci	Komunikacja radiowa 433 MHz					
<b>0</b>	Panel obsługi			microUSB ↔ USB	Panel operatorski/ Interfejs użytkownika	
	Moduł: Radio 433 MHz	Port szeregowy UART	Moduł: Adafruit Feather M0	Port szeregowy UART	Komputer	
	Nawiązanie łączności i wymiana danych wewnątrz sieci radiowej	Transmisja danych ↔	Moduł pośredniczący	Transmisja danych ↔	<ol style="list-style-type: none"> <li>1. Weryfikacja parametrów symulacyjnych z fizycznym systemem</li> <li>2. Możliwe sterowanie lotem UAS</li> <li>3. Wizualizacja pracy UAS</li> </ol>	
<b>1</b>	Bezzałogowe urządzenie latające – 1					
	Moduł: STM F7 Procesor 1		Port szeregowy UART	Moduł: ESP-32 Procesor 2	Port szeregowy UART	Moduł: Radio 433 MHz
	<b>Priorytet</b>	<b>Stabilizacja</b>	Transmisja danych ↔	Algorytm lotu automatycznego i autonomicznego	Transmisja danych ↔	Nawiązanie łączności i wymiana danych wewnątrz sieci radiowej
	1	Osi $x, y, z$				
	2	Wysokości				
3	Pozycji					
4	Prędkości					
<b>2</b>	Bezzałogowe urządzenie latające – 2					
<b>3</b>	Bezzałogowe urządzenie latające – 3					
<b>4</b>	Bezzałogowe urządzenie latające – 4					
<b>5</b>	Bezzałogowe urządzenie latające – 5					

Rys. 3.8. Struktura Bezzałogowego Systemu Latającego

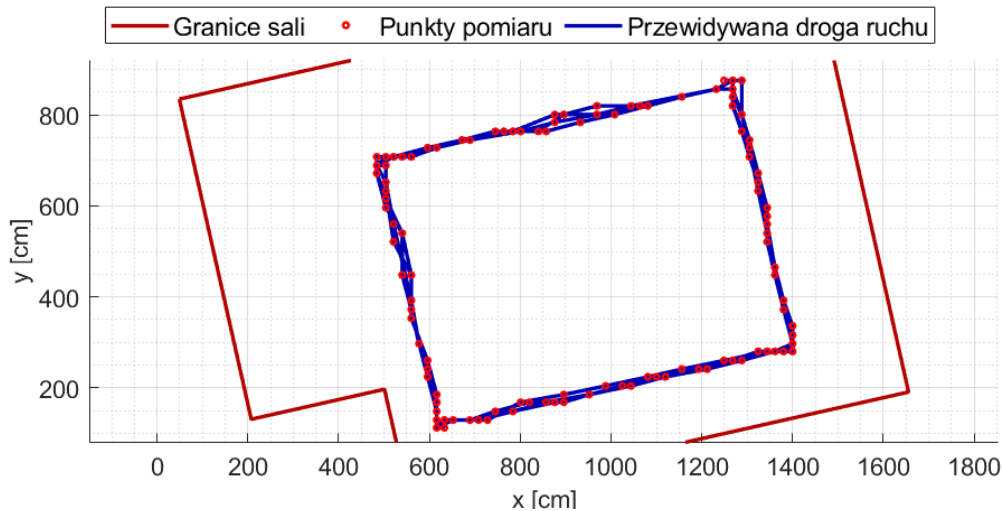
### 3.3 Systemy pozycjonowania wykorzystane w urządzeniu

Zadaniem systemu pozycjonowania było dostarczenie UAS informacji z możliwie jak największą dokładnością. Zarówno nawigacja satelitarna, jak i lokalny system *Marvelmind* dostarczały także informacji o prędkości urządzenia.

Na wybór rozwiązania firmy *Marvelmind* duży wpływ miała cena zestawu w stosunku do deklarowanych przez producenta parametrów. W końcowej wersji 11 punktów nawigacyjnych

(6 stacjonarnych oraz 5 mobilnych) zamknęła się w kwocie 4 tys. zł. Porównywalne systemy oparte na systemach wizyjnych kosztowały kilkanaście razy więcej.

Na rysunku 3.9 pokazano zarejestrowaną drogę przy przemieszczeniu odbiornika po obrysach prostokąta o wymiarach  $8 \times 6$  m z prędkością 1 m/s, gdy w system wpięto pięć mobilnych punktów nawigacji. Czerwone punkty wskazują punkt pomiaru. Ze względu na stosowany protokół NMEA 0183 (wymuszony przez kontroler lotu) rozdzielczość pomiaru wynosiła około 20 cm.



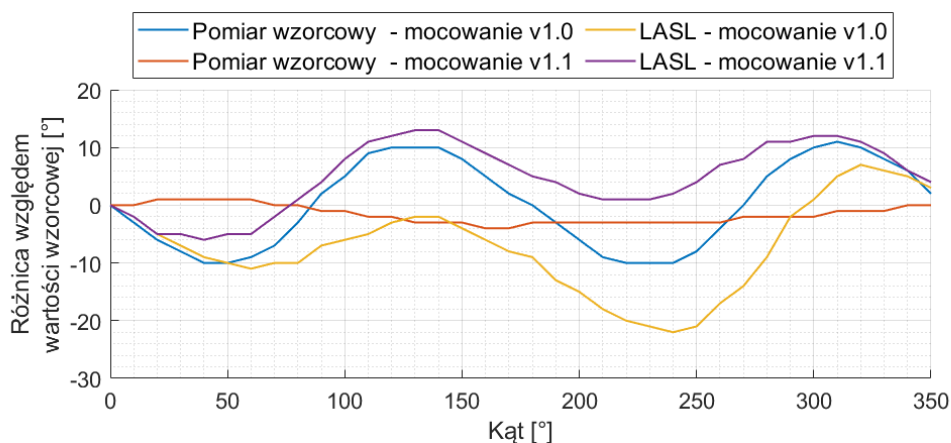
**Rys. 3.9.** Pomiar przemieszczenia 2D z wykorzystaniem systemu nawigacji Marvelmind wzdłuż prostokąta o wymiarach  $8 \times 6$  metrów dla: **a)** jednego urządzenia, **b)** pięciu urządzeń

Podczas eksperymentów możliwe było określenie rzeczywistej częstotliwości odświeżania pozycji, poprzez rejestrację zmian wartości mierzonej (wzdłuż jednej osi), w trakcie ruchu. Na podstawie uzyskanych wyników ustalono, iż czas aktualizacji pozycji (dla jednego MB) mieści się średnio w przedziale 0,1 – 0,2 sekundy, co jest nieznaczenie gorsze od deklaracji producenta. Dla pięciu MB uzyskiwano aktualizację pozycji co 0,7 – 0,8 s.

### 3.4 Określenie zwrotu w układzie współrzędnych

Urządzenie do lotu automatycznego/autonomicznego poza pozycją, wymaga także informacji o zwrocie/kierunku urządzenia w układzie globalnym R3<sub>1</sub>. Najpopularniejszym czujnikiem jest kompas, który wykorzystując magnetometr mierzy stałe pole magnetyczne ziemi. Na jego podstawie wyznacza się kąt obrotu czujnika względem północy. Niestety jego stosowanie nie zawsze jest możliwe. W budynkach bądź w pobliżu urządzeń elektrycznych pomiar może zostać obciążony dużym błędem.

Pierwszym zbadanym czujnikiem był trzyosiowy magnetometr HMC5883L. Wadą magnetometru jest jego podatność na obecność w pobliżu elementów ferromagnetycznych (konstrukcje stalowe, zbrojenia w ścianach) oraz urządzeń generujących pole elektromagnetyczne (instalacje elektryczne, monitory, komputery, kamery). Dlatego zaleca się niestosowanie w budynkach tego typu rozwiązania. Na rysunku 3.10 pokazano zachowanie magnetometru w centralnej części laboratorium. Analiza wyników wskazuje jednoznacznie pogorszenie dokładności w porównaniu do pomiarów wzorcowych, wykonanych poza zabudowaniami, z dala od potencjalnych źródeł mogących zniekształcić pomiar. Zauważono, że czujnik w różnych punktach laboratorium nie zachowuje się tak samo, a rozbieżności sięgają 20°.



**Rys. 3.10.** Rozkład błędów w zależności od rodzaju mocowania czujnika oraz kąta obrotu na dewiację wyniku (magnetometr HMC5883L)

Ze względu na brak działających, alternatywnych rozwiązań zdecydowano się na wykorzystywanie magnetometru, jednak w ograniczonym zakresie. W algorytmie sterowania założono, że urządzenie utrzymuje stałą wartość parametru kąta obrotu wzdłuż osi z. W celu korekcji wskazań czujnika zastosowano funkcję aproksymującą w postaci wielomianu drugiego rzędu.

## 4. Decentralizacja sieci radiowej

Jednym z istotnych zagadnień niniejszej pracy było wykonanie zdecentralizowanej sieci radiowej. Rozumie się przez to nieprzerwaną, o ile to możliwe, komunikację grupy dronów bez względu na ewentualne awarie pojedynczych jednostek. Aby zapewnić decentralizację bezzałogowego systemu latającego trzeba ją rozważyć na dwóch płaszczyznach. Pierwsza zakłada sterowanie ruchem i pozycją UAV – zagadnienie to zostało opisane w rozdziałach 8 – 9. Kolejnym aspektem jest wymiana informacji pomiędzy urządzeniami.

### 4.1 Założenia strukturalne sieci radiowej

Strukturę pracy sieci zaprojektowano w oparciu o następujące założenia:

- użytkownicy sieci łączą się bezpośrednio, dynamicznie i niehierarchicznie,
- możliwe jest wejście i wyjście z sieci w każdym momencie,
- możliwe jest wejście do sieci nowego użytkownika, a także jej opuszczenie w sposób niezagrażający stabilności wymiany informacji pomiędzy pozostałymi użytkownikami,
- maksymalna liczba użytkowników jest stała i przy starcie systemu zadeklarowana,
- liczba członków w sieci może się dynamicznie zmieniać,
- transmisja danych jest szyfrowana, a wejście do sieci jest możliwe tylko przy znajomości 16 - bajtowego klucza,
- niedopuszczalna jest przerwa w transmisji danych większa niż 1 sekunda,
- każdy moduł pracujący wewnątrz sieci musi posiadać status obserwatora bądź członka sieci.

### 4.2 Zasada działania sieci radiowej

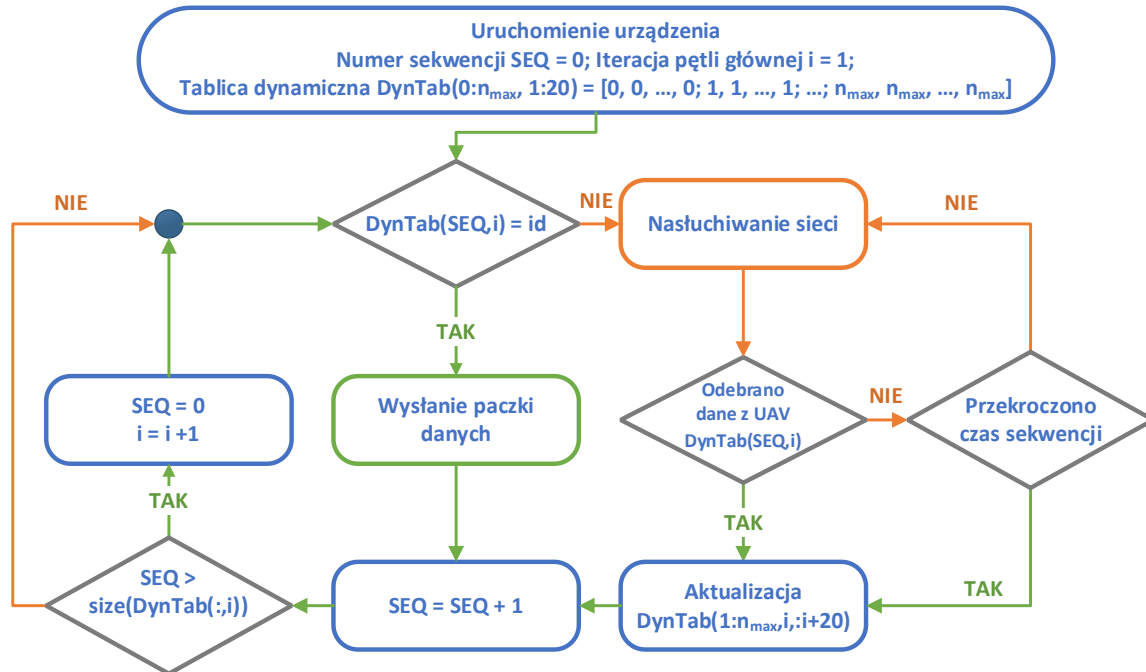
Na rysunku 4.1 pokazano schemat blokowy pracy sieci radiowej. Każde urządzenie tuż po uruchomieniu tworzy następujące zmienne:

- numer sekwencji:  $SEQ = 1$ ,
- iteracja pętli:  $i = 1$ ,

- tablica dynamiczna dla następnych 20 iteracji:

$$\text{DynTab}(0:n, 1:20) = [0, 0, \dots, 0; 1, 1, \dots, 1; \dots; n, n, \dots, n].$$

Liczba wierszy *DynTab* jest uzależniona od maksymalnej liczby członków sieci  $n_{\max}$ , natomiast liczba kolumn odwzoruje opóźnienie reakcji sieci na ewentualny zanik łączności z dowolnym użytkownikiem. W niniejszej pracy początkowy rozmiar wynosi 6 (pięć UAV + jedno dla panelu operatorskiego)  $\times$  20. Wspomniane wcześniej parametry są tworzone przez każdego uczestnika sieci indywidualnie i mogą na pewnych etapach pracy systemu się od siebie różnić.



Rys. 4.1. Schemat blokowy komunikacji wewnątrz sieci radiowej

### 4.3 Testy sieci radiowej

W ramach pracy sprawdzono użyteczność trzech różnych podejść zarządzania rozmiarem sieci radiowej (parametr  $n_{\max}$ ). Pierwsza zakładała stałą wartość równą maksymalnej możliwej liczbie członków sieci. W ramach pracy przyjęto ten parametr równy sześć (pięć UAV oraz panel obsługi). Przyjęcie stałej wartości wiąże się z problemem oczekiwania na pakiet danych od urządzeń które w chwili obecnej nie pracują. Czas oczekiwania na paczkę danych przyjęto na poziomie 30 ms, tzn. jeśli jedno urządzenie w danej chwili nie pracuje, każdy inny członek sieci niepotrzebnie oczekuje w każdym przejściu na wymianę paczki danych 30 ms. Przy dwóch nieobecnych członkach czas ten wynosi 60 ms, i analogicznie wzrasta przy większej liczbie nieobecnych UAV. W celu eliminacji takiego zachowania zaproponowano algorytm dynamicznie zmieniający wielkość sieci. Parametr maksymalnej liczby użytkowników w sieci ( $n_{\max}$ ) wyznaczano zgodnie ze wzorem  $n_{\max} = n + 1$ , gdzie  $n$  jest aktualną liczbą użytkowników w sieci. Zaletą podejścia było dodanie nowego urządzenia bezpośrednio po otrzymaniu paczki danych. Czas inicjalizacji wynosił w takim przypadku maksymalnie 500 ms. W tabeli 4.1 rozwiązanie to znajduje się pod numerem drugim.

Dla rozwiązania trzeciego, parametr  $n_{\max}$  był równy wartości  $n$ . Ze względu na brak oczekiwania przez system na ewentualnego nowego członka sieci, uzyskano większe częstotliwości w porównaniu z rozwiązaniem drugim. Wadą podejścia była dłuższa stabilizacja systemu po zmianie wielkości sieci (usunięcie bądź dodanie członka sieci). Częstotliwość pracy sieci radiowej stabilizowała się po około 2 – 3 sekundach. W tabeli 4.1 umieszczono zmierzoną częstotliwość aktualizacji pakietu danych każdego urządzenia w sieci UAV.

**Tabela. 4.1.** Zmierzona częstotliwość aktualizacji paczki danych każdego w sieci członka w zależności od stosowanego rozwiązania

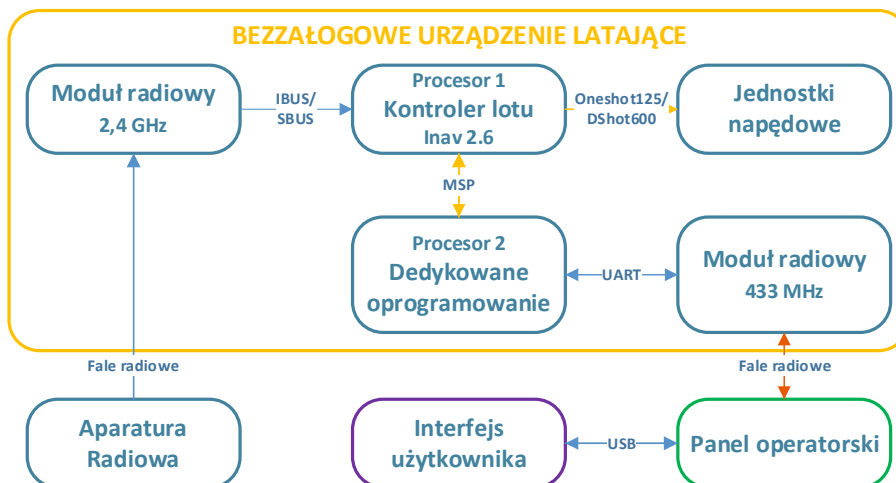
Rozwiązanie			Liczba użytkowników w sieci				
			n = 2	n = 3	n = 4	n = 5	n = 6
1.	Stały rozmiar sieci $n_{\max} = \text{const}$	$n_{\max} = 2$	64 Hz	-	-	-	-
		$n_{\max} = 3$	44 Hz	42,5 Hz	-	-	-
		$n_{\max} = 4$	40 Hz	27,5 Hz	33 Hz	-	-
		$n_{\max} = 5$	38 Hz	23,5 Hz	21,7 Hz	26 Hz	-
		$n_{\max} = 6$	36 Hz	22 Hz	18,3 Hz	18 Hz	22 Hz
2.	Dynamiczny rozmiar sieci	$n_{\max} = n + 1$	44 Hz	27,5 Hz	21,7 Hz	18 Hz	22 Hz
$n_{\max} = n$		64 Hz	42,5 Hz	33 Hz	26 Hz	22 Hz	

## 5. Sterowanie pojedynczym dronem (UAV) oraz grupą dronów (UAS)

Stosowane w pracy czterosilnikowe bezzałogowe urządzenia latające należą do grupy urządzeń, które bez dedykowanego układu sterowania nie potrafiłyby latać. Zgodnie ze schematem przedstawionym w rozdziale trzecim, jednostka obliczeniowa składa się z dwóch modułów. Kontroler lotu wraz z oprogramowaniem Inav 2.6 zawiera algorytmy pozwalające na sterowanie przez operatora UAV za pośrednictwem dedykowanej aparatury radiowej. Realizacja postawionych w rozprawie celów wymagała dodania drugiego modułu ESP-32, który zastępując operatora przejmuje sterowanie torem lotu oraz komunikacją w UAS w czasie rzeczywistym.

### 5.1 Wybór sposobu sterowania UAV

Na rysunku 5.1 pokazano schemat wymiany danych. Aparaturę wraz z jej modułem radiowym pozostawiono, aby w razie awarii bądź niekontrolowanego zachowania urządzenia, możliwe było przejście na tryb ręczny i wylądowanie urządzeniem.

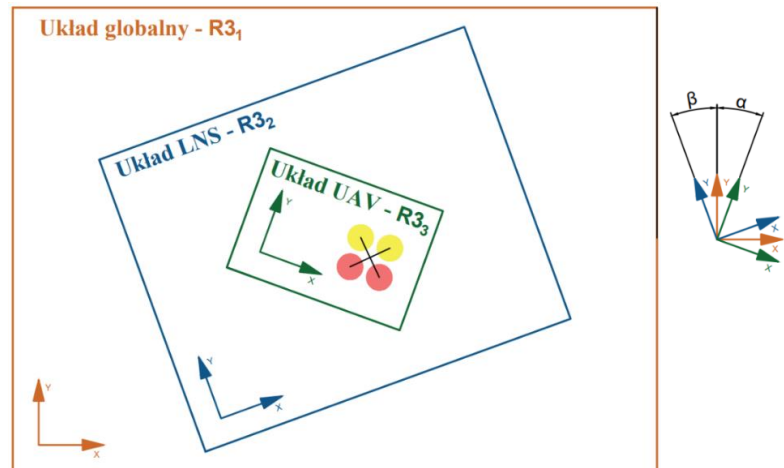
**Rys. 5.1.** Schemat wymiany danych stosowany w pracy

### 5.2 Układy współrzędnych

System pracuje z wykorzystaniem trzech układów współrzędnych przesuniętych względem siebie o kąty  $\alpha$  i  $\beta$  tak jak pokazano na rysunku 5.2.



1. Globalny układ współrzędnych  $R_{3_1}$  – GPS, magnetometr
2. Lokalny układ współrzędnych  $R_{3_2}$  – Nawigacja Marvelmind
3. Wewnątrz UAV –  $R_{3_3}$



Rys. 5.2. Układy współrzędnych w UAS

Pierwszy układ współrzędnych  $R_{3_1}$  jest globalny i stanowi podstawę w sterowaniu. Pozostałe układy są przeliczane względem niego. W globalnym układzie  $R_{3_1}$  pracuje system nawigacji GPS oraz magnetometr. Drugi układ  $R_{3_2}$  został wymuszony przez zastosowany system nawigacji lokalnej Marvelmind. Niestety system ten w podstawowej konfiguracji nie zwraca zwrotu/kierunku urządzenia. Do jego wyznaczenia stosowano nadal magnetometr, a cały układ obrócono o kąt  $\beta$  wokół osi  $z$ , tak aby osie  $x$ ,  $y$ ,  $z$  obu układów były względem siebie równoległe. Rotację wykonuje się raz przy starcie systemu – następnie pozycja jest automatycznie przeliczana względem kąta obrotu. Trzeci układ  $R_{3_3}$  znajduje się wewnątrz każdego urządzenia i jest wykorzystywany do wyznaczenia wektora kierunkowego wymaganego do sterowania. UAV będąc w ruchu, wykonując manewr, obraca się o kąt  $\alpha$ . Układ ten jest stale w ruchu, stąd wektor kierunkowy, pomimo że punkt celu może pozostać bez zmian, może zmieniać swoją wartość. Przyjęto także, że urządzenie zawsze znajduje się w początku układu współrzędnych, a współrzędne punktu celu w układzie  $R_{3_3}$  stanowią faktyczną odległość urządzenia od punktu celu w układzie  $R_{3_1}$ .

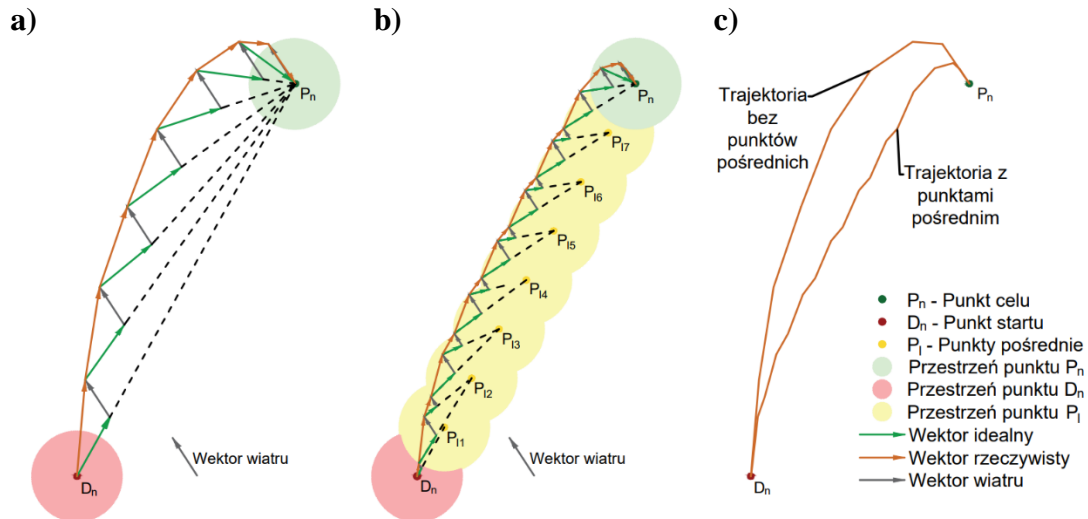
### 5.3 Generowanie trajektorii

Przed startem urządzenia konieczne jest określenie trajektorii lotu. Operator systemu może wgrać trajektorię wpisując ją ręcznie w postaci współrzędnych  $[x, y, z]$  punktów przelotowych bądź wykorzystać napisane oprogramowanie. W ramach pracy założono trzy rodzaje punktów przelotowych. Urządzenie rozpoczynając lot startuje z punktu startu i zmierza w kierunku punktu celu. Jeśli odległość pomiędzy tymi punktami jest większa niż dwukrotność dokładności stosowanego systemu nawigacyjnego generowane są punkty pośrednie. Stosowanie ich znacząco poprawia dokładność lotu, ponieważ w jej trakcie UAV korzystają z utworzonego wektora, gdzie punkt zaczepienia jest jego pozycją, a zwrot określa punkt celu. Gdyby lot urządzenia został zakłócony przez czynniki zewnętrzne (np. wiatr), przebyta droga mogłaby przypominać łuk (rys. 5.3a). W przypadku lotu grupowego takie podejście może spowodować kolizję. Utworzenie kilku punktów pomiędzy  $D_{1:n} - P_{1:n}$  zwiększa kąt pomiędzy wektorem idealnym i rzeczywistym, co w następstwie zwiększa wartości sterowania, a urządzenie leci z większą dokładnością (rys. 5.3b). UAV znajdując się w zasięgu punktu pośredniego (strefa żółta) zalicza je i zmierza do następnego. Porównując oba przypadki, lepszą precyzję lotu uzyskuje się z wykorzystaniem punktów pośrednich (rys. 5.3c).

Do obsługi systemu napisano dedykowane oprogramowanie, pozwalające na wygenerowanie trajektorii, wysłanie jej do urządzenia oraz nadzór nad urządzeniem w trakcie lotu. Cały system składa się z minimum jednego urządzenia latającego, systemu komunikacji oraz komputera klasy PC wraz z dedykowanym oprogramowaniem. W pracy założono, że użytkownik powinien wpisać charakterystyczne punkty celu oraz zdefiniować kształt pomiędzy nimi. Trajektorium pomiędzy punktami może przyjmować kształt prostej, łuku bądź helisy. Jeżeli odległość pomiędzy punktami

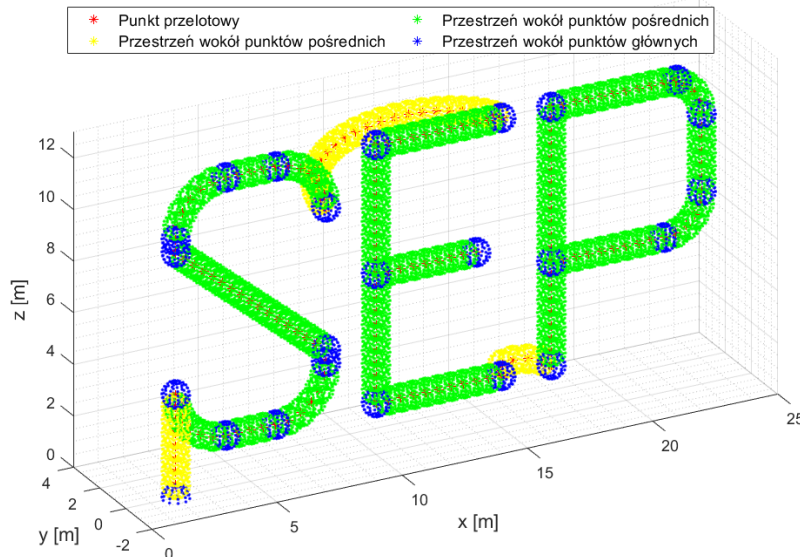
głównymi będzie większa niż zadeklarowana dokładność, program sam wyliczy punkty pośrednie w taki sposób, aby zdefiniowany kształt został zachowany.

Trajektorię dla bezzałogowego systemu latającego utworzono analogicznie jak dla pojedynczego urządzenia. Podstawowymi kształtami były odcinek, okrąg bądź jego wycinek i helisa. Podstawowa trajektoria zakładała również podanie punktów przelotowych oraz punktu celu.



**Rys. 5.3.** Idea wykorzystania punktów pośrednich w przestrzeni 2D dla jednej pary  $D - P$  w warunkach idealnych (wiatr jest stały w czasie): **a)** bez punktów pośrednich, **b)** z punktami pośrednimi, **c)** porównanie trajektorii z przypadku a oraz b

Na rys. 5.4 pokazano złożenie kilku kształtów podstawowych. Miejsce uzbrojenia silników urządzenia definiuje punkt (2,0,0) w globalnym układzie współrzędnych  $R_{31}$ . Względem tego punktu były każdorazowo wyliczane współrzędne głównych i pośrednich punktów przelotowych. Program zapisuje trajektorię do pliku *.txt* oraz wysyła do urządzenia. Kolorem żółtym i zielonym zaznaczono tor lotu.



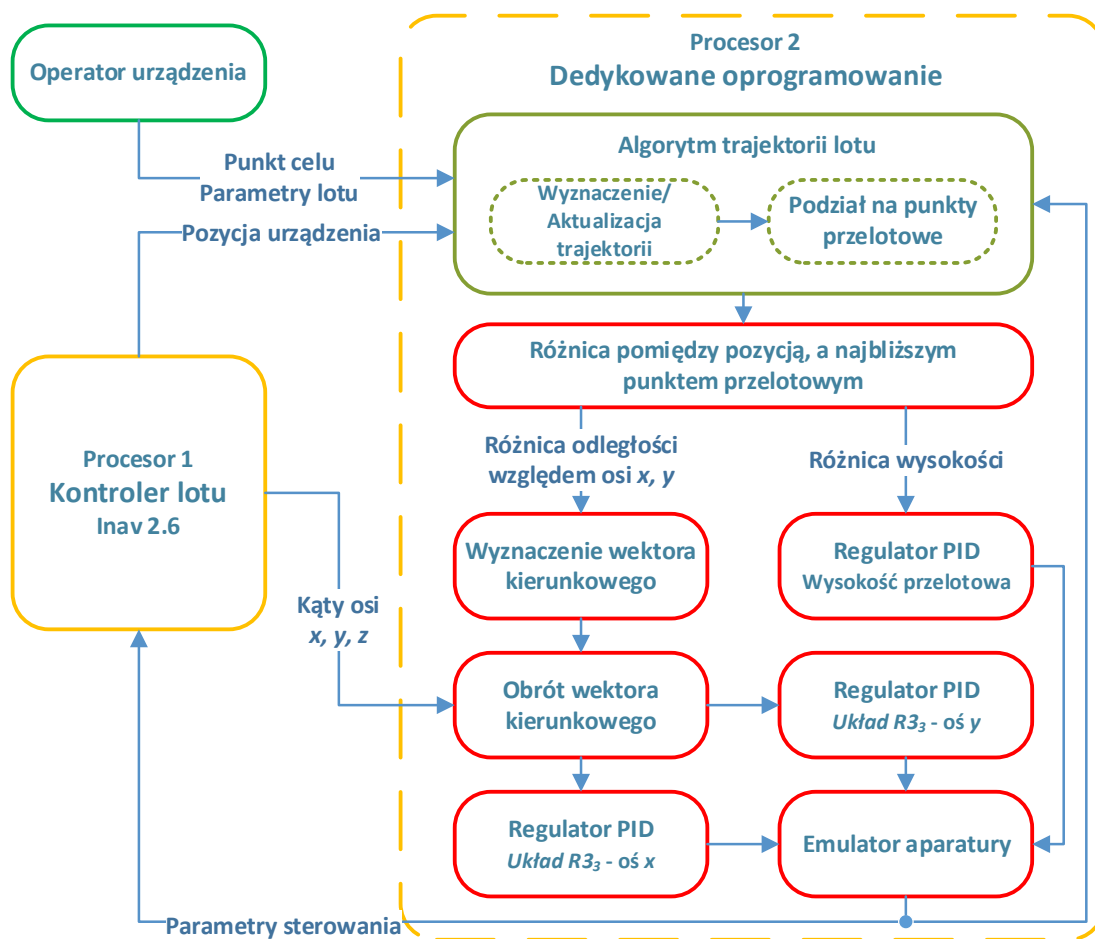
**Rys. 5.4.** Wygenerowana trajektoria w kształcie akronimu Stowarzyszenia Elektryków Polskich

## 5.4 Algorytm sterowania

### 5.3.1. Bezzałogowe urządzenie latające

Na rysunku 5.5 pokazano schemat systemu sterowania pojedynczego urządzenia. Kolorem zielonym oznaczono blok operatora, którego zadaniem jest uruchomienie urządzenia oraz zdefiniowania zadania poprzez deklarację punktu celu i parametrów lotu. Operator, korzystając z panelu obsługi może w każdej chwili wysłać instrukcję o lądowaniu bądź awaryjnie (w ostateczności) rozbroić silniki i doprowadzić do jego upadku. Kolorem żółtym oznaczono elementy znajdujące się na urządzeniu. Napisane oprogramowanie (na procesorze drugim) w każdej iteracji pętli pobiera instrukcje (uzbroić/rozbroić silniki, lot bądź lądowanie) od operatora, oraz z kontrolera lotu pozycję i kąty osi  $x$ ,  $y$ ,  $z$ , generując na ich podstawie trajektorię lotu.

Znając pozycję urządzenia, kąty osi oraz współrzędne punktu celu można wyznaczyć parametry sterowania. Pierwszym etapem jest policzenie odległości urządzenia do najbliższego punktu przelotowego w przestrzeni  $R^3$  względem osi  $x$ ,  $y$ ,  $z$ . Następnie algorytm idzie dwutorowo. Parametr wysokości jest podawany bezpośrednio na jeden z trzech regulatorów PID. Parametry osi  $x$ ,  $y$  muszą zostać dopasowane do układu współrzędnych  $R^3$ , gdyż urządzenie może być obrócone względem układu  $R^3$  o kąt  $\alpha$ . Współrzędne zwrotu, wektora kierunkowego w  $R^3$  stanowią wartość wejściową dwóch regulatorów odpowiedzialnych za przemieszczenie urządzenia względem osi  $x$ ,  $y$ . Ostatnim etapem było dopasowanie sygnałów trzech regulatorów PID do wartości sterowania. Napisany emulator aparatury przetwarza wyznaczone przez regulatory wartości sterujące i wysyła do kontrolera lotu. Opisany cykl powtarzany w każdej iteracji pracy kontrolera.



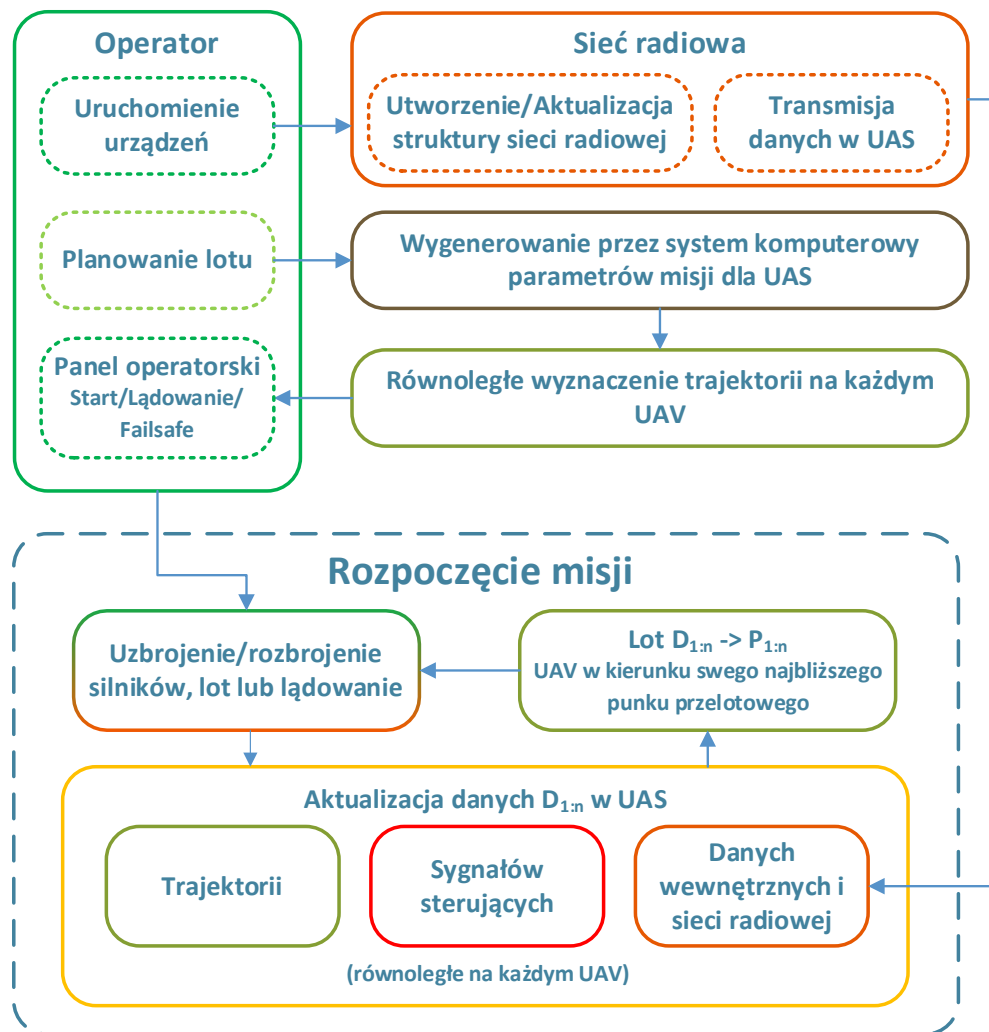
Rys. 5.5. Schemat ideowy układu sterowania dla bezzałogowego urządzenia latającego

### 5.3.2. Bezzałogowy system latający

Proces planowania trajektorii zakładał wygenerowanie w odpowiedniej kolejności punktów przelotowych tworząc dla każdego UAV, wirtualny „tunel”, w którym system powinien się poruszać. Na rysunku 5.6 pokazano uproszczony schemat blokowy z uwzględnieniem instrukcji warunkowych UAV/UAS. Ideą algorytmu jest aby każde urządzenie dążyło do swojego następnego punktu przelotowego i jeśli znajdzie się w jego pobliżu z zadeklarowaną dokładnością, system uzna punkt za zaliczony i zaktualizuje go o następny, aż do uzyskania ostatniego punktu celu. System w swojej pętli decyzyjnej sprawdza także czy formacja wymaga zmiany bądź czy punkt celu się nie zmienił.

Każde urządzenie bezpośrednio po podłączeniu do zasilania tworzy swoją własną sieć radiową, szukając innych użytkowników. Jeśli dojdzie do znalezienia innego UAV, sieci są scalane i komunikacja od tego momentu jest możliwa.

Planowanie lotów o prostej trajektorii jest możliwe poprzez wysłanie punktu celu bezpośrednio do urządzeń. W ramach pracy napisano oprogramowanie, które pozwala na stworzenie bardziej zaawansowanych trajektorii, przykładowo w kształcie wybranych figur bądź liter. Żółty blok na rysunku 5.6 reprezentuje operacje wykonane na urządzeniu. Zaletą systemu jest brak zależności bądź podziału wykonywanych obliczeń w sieci. Każde urządzenie znając pozycję innych urządzeń jest w stanie niezależnie wyliczyć jego trajektorię i dokonać predykcji jego pozycji.



Rys. 5.6. Schemat ideowy pracy UAS

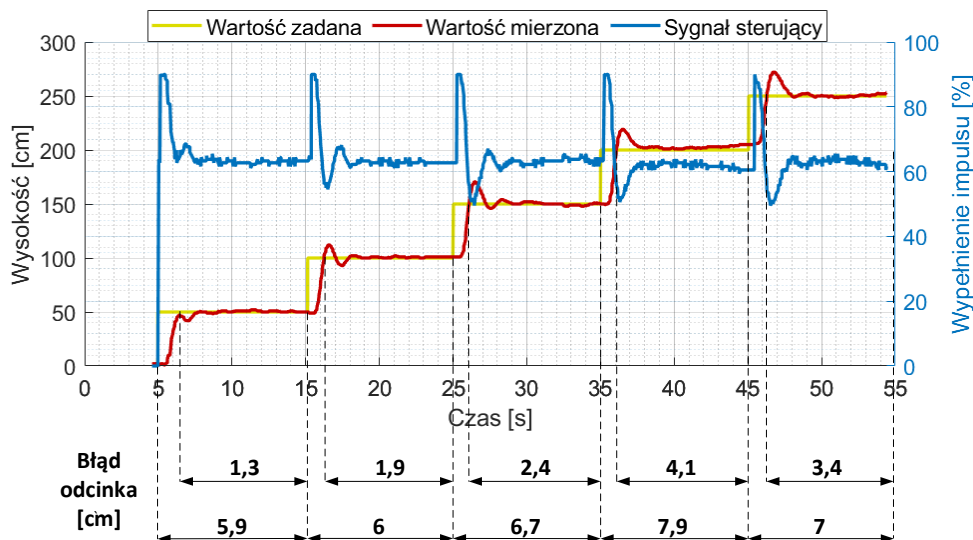
## 6. Testy pojedynczego UAV

W ramach pracy przeprowadzono szereg testów w celu określenia właściwości UAV. Jeden z nich zakładał pomiar wskazania magnetometru oraz parametru RSSI w zależności od rodzaju oraz umiejscowienia anteny radiowej. Badania przeprowadzono w wolnej przestrzeni odzwierciedlając normalne warunki pracy urządzenia oraz w komorze bezodbićowej w celu wyeliminowania wpływu fal radiowych (GSM, GPS, Wi-Fi, Radio) na badane podzespoły. Zbadano także reakcję komunikacji radiowej i magnetometru na uruchomienie, pracę silników oraz zachowanie przy linii najwyższego napięcia. Z zasady działania kompasu (pomiar pola magnetycznego) oraz modułu radiowego (pomiar sygnału fal elektromagnetycznych) wynika, że powinny być one szczególnie podatne na wirujące pole magnetyczne generowane właśnie przez silniki. Dodatkowo (w stosowanych urządzeniach) podczas startu prąd może skokowo wzrosnąć nawet do 100 A, doprowadzając do wahań napięcia. Mogą one zakłócać mierzone sygnały na antenie przez moduł radiowy.

W niniejszym rozdziale przedstawiono wyniki pomiarowe dla autorskiego układu sterowania zaimplementowanego wewnątrz jednego urządzenia z wykorzystaniem systemu nawigacji lokalnej Marvelmind.

### 6.1 Wzniesienie

Na rysunku 6.1 pokazano reakcję UAV na skokowe zwiększanie wartości zadanej o 50 cm do wartości 250 cm. Dla całego wykresu błąd wynosił 6,8 cm. Na pogorszenie wyniku wpływał czas osiągnięcia wartości zadanej oraz lekkie przeregulowanie. Po uzyskaniu wartości zadanej błąd systemu, nawet na wysokościach w okolicach czterech metrów nie przekraczał pięciu centymetrów. Celem doświadczenia była weryfikacja stabilności pracy systemu. Analiza wykresu wskazuje, że system, pod kątem uzyskania deklarowanej wysokości działa stabilnie.



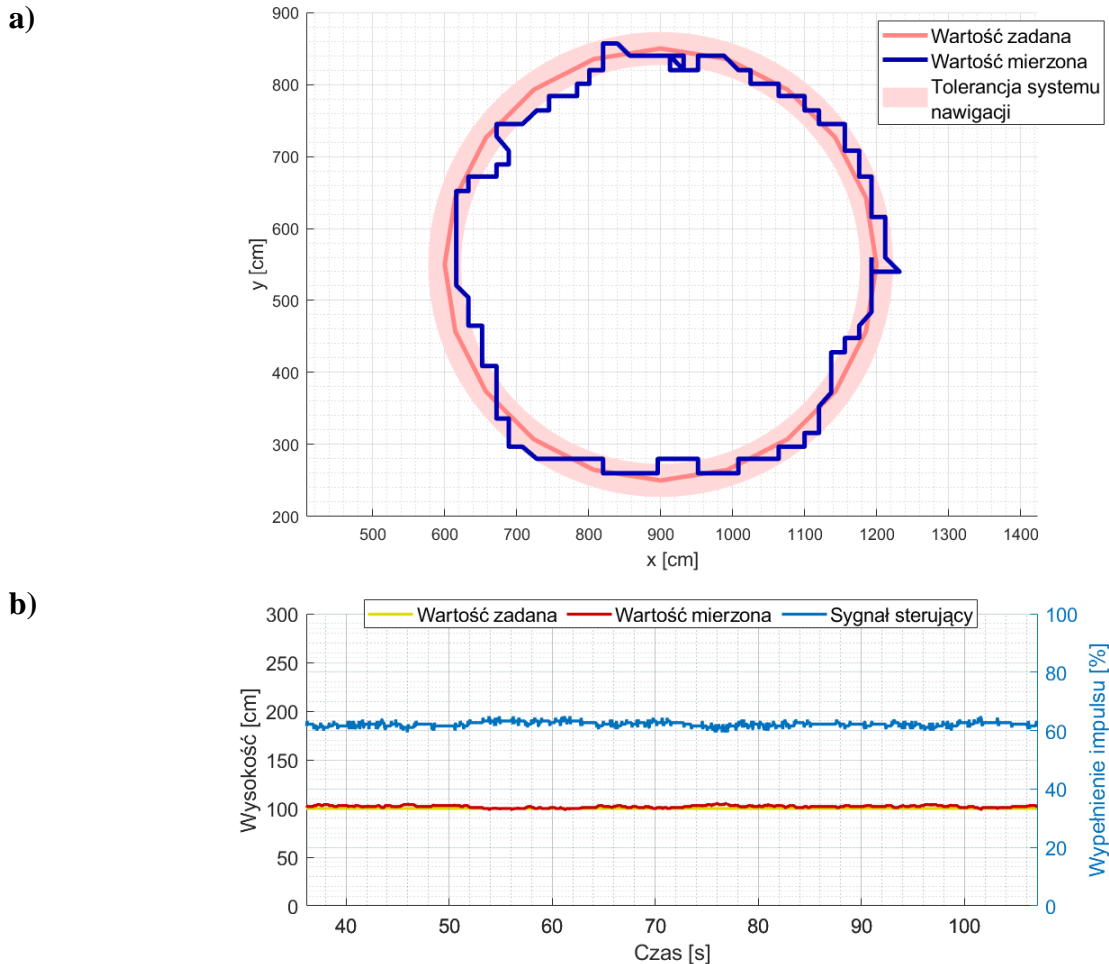
Rys. 6.1. Skokowe zwiększanie wartości zadanej do poziomu 250 cm

### 6.2 Lot po punktach przelotowych

Kolejnym etapem były badania związane z przemieszczeniem urządzenia. Ze względów bezpieczeństwa, każde urządzenie, aby mogło przemieścić się w którąkolwiek stronę, musi uzyskać wysokość minimum 50 cm. Takie działanie miało na celu uniknięcie zbyt dużego kąta startu oraz manewrów na niskiej wysokości, które mogą prowadzić do uszkodzenia UAV. Wszystkie niżej zaprezentowane wyniki wykonano na wysokości 100 cm.

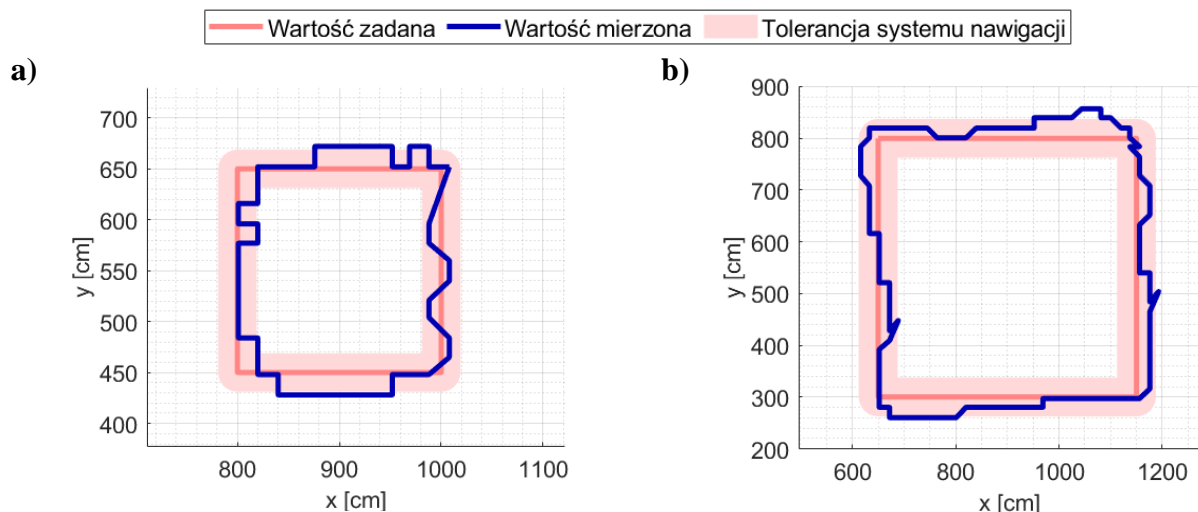


Na rysunku 6.2a pokazano zarejestrowaną drogę jednego urządzenia, które poruszało się, po uzyskaniu 100 cm, po okręgu o średnicy 600 cm. Różowa gruba linia określa tolerancję systemu nawigacji. Analiza wyników wskazuje, że urządzenie poruszało się płynnie wzdłuż wcześniej zaplanowanej trajektorii. Układ sterowania korygował pozycję z częstotliwością 21 Hz. System pozycjonowania aktualizował pozycje co 5 – 10 Hz. Na wykresie 6.2b pokazano zarejestrowaną przez urządzenie wysokość w trakcie lotu po okręgu. Wyniki wskazują, że nawet w trakcie lotu system jest w stanie stabilnie utrzymać swoją deklarowaną wysokość.



**Rys. 6.2.** Lot po okręgu o średnicy 600 cm: **a)** przemieszczenie 2D, **b)** parametr wysokości wraz z wartością sterowania,

Na rysunku 6.3a pokazano lot po krawędzi kwadratu o boku 200 cm. Ruch po tak małych figurach, zawierających z swoim kształcie załamanie trajektorii (wierzchołek) był często problematyczny ze względu na niską częstotliwość odświeżenia systemu nawigacji. Kolejny rysunek 6.3b przedstawia lot po obrzeżach kwadratu o długości boku 500 cm. Na pokazanych rysunkach urządzenia poruszały się zgodnie z zaplanowanym torem lotu, rzadko wychodząc poza tolerancję systemu nawigacji.



Rys. 6.3. Wybrane formacje lotu: a) kwadrat o boku 200 cm, b) kwadrat o boku 500 cm

## 7. Laboratorium Autonomicznych Systemów Latających

Istotną częścią pracy doktorskiej było przygotowanie zamkniętego laboratorium do badania bezzałogowych systemów latających. W kolejnych podrozdziałach opisano poszczególne jego elementy oraz trudności związane z jego wykonaniem.

### 7.1 Konstrukcja sali

Pomieszczenie, w którym badano UAS nazwano *Laboratorium Autonomicznych Systemów Latających* (w skrócie *LASL*) na wzór laboratorium *Flying Machine Arena (FMA)* z ETH Zurich [10]. Autorska konstrukcja składa się z plandek oraz siatek ochronnych, mających na celu łapanie urządzeń. Należy nadmienić, że w pierwszej fazie testów urządzenia były szczególnie podatne na uderzenia bądź kolizje. Rysunek 7.1 przedstawia zdjęcia wnętrza sali laboratoryjnej. Możliwa przestrzeń ruchu urządzeń wynosiła 11×15×5 metra [szerokość, długość, wysokość]. W rogu laboratorium umieszczono *klatkę*, w której znajdowało się centrum obsługi całego systemu.



Rys. 7.1. Wnętrze Sali laboratoryjnej – dwa słupy nawigacyjne oraz klatka bezpieczeństwa

## 7.2 System nawigacji

Na podstawie przeglądu dostępnych rozwiązań, głównie ze względów ekonomicznych oraz deklarowanej przez producenta dokładności wybrano rozwiązanie firmy Marvelmind. Zamontowany w laboratorium system składał się z:

- Dwóch map podrzędnych:
  - pierwsza z priorytetem na parametry pozycji 2D (osi  $x$ ,  $y$ ) zawiera 4 stacjonarne nadajniki zamontowane na wysokości 5 metrów,
  - druga do wsparcia systemu, zwiększająca dokładność parametru osi  $z$  – mierzonej wysokości, zawierająca również 4 stacjonarne nadajniki. Dwa z nich są współdzielone zatem system zawiera 6 stacjonarnych nadajników.
- Maksymalnie pięciu mobilnych nadajników/odbiorników nawigacji. Wersje na urządzeniach zawierały czujnik IMU (9DOF: akcelerometr 3D, żyroskop 3D i magnetometr 3D) wykorzystywany w celu zwiększenia dokładności wskazań.
- Routera wpiętego do komputera wraz z oprogramowaniem.

## 7.3 Wyznaczenie zwrotu urządzenia

Jednym z problemów wewnątrz LASL było wyznaczenie zwrotu urządzenia. Ze względu na brak innych możliwości zdecydowano się na wykorzystanie magnetometru. W ramach pracy zbadano kilka różnych typów magnetometrów znajdujących się obecnie na rynku. W celu weryfikacji ich wskazań, LASL podzielono na 15 mniejszych stref o wielkości  $3 \times 4$  metry. Wewnątrz każdej z piętnastu stref wykonano pomiar, gdy urządzenie znajdowało się na powierzchni ziemi, na wysokości jednego oraz dwóch metrów. Każdy pomiar wykonano dla czterech kątów wzorcowych: 45, 135, 225 i 315°.

**Tabela 7.1.** Maksymalne rozbieżności  $|max - min|$  wskazania badanych magnetometrów w wybranych strefach laboratorium w zależności od wysokości

Czujnik	Strefa	1 – 15	4 – 15	4 – 12
HMC5883L	Podłoga	22,5°	15°	13,5°
	1 metr	19,5°	14°	13°
	2 metry	17°	15°	10,5°
LIS3MDL	Podłoga	29,5°	17,5°	16,75°
	1 metr	22,25°	14,75°	15°
	2 metry	21,75°	19,25°	14,25°

Wstępna analiza wyników wskazywała na duże rozbieżności wskazań pomiędzy wybranymi strefami (tabela 7.1), np. gdy urządzenie znajdowało się przy powierzchni wynosiła dla LIS3MDL 29,5°, a dla HMC5883L 22,5°. Takie wyniki nie pozwalały na skuteczne sterowanie urządzeniami. Na wysokości jednego oraz dwóch metrów wskazania były nieco dokładniejsze.

Na podstawie przeprowadzonych badań utworzono 2-wymiarową funkcję aproksymującą wielomianu drugiego rzędu korygującą pomiar. Przemnażając wskazanie czujnika przez współczynnik korygujący uzyskuje się wskazanie bliskie rzeczywistości. Funkcję korygującą wyznaczono celowo dla układu  $R3_2$  tzn. względem układu nieprzesuniętego o kąt  $\beta$ . Ostatecznie, każde urządzenie znajdujące swoją pozycję względem  $R3_1$  przelicza ją zgodnie ze wzorem 7.1 – 7.2 na współrzędne układu  $R3_2$ . Następnie koryguje swoje wskazanie w zależności od wysokości według wzoru 7.3 (wysokość 1) albo 7.4 (wysokość 2 m).

$$xR3_2 = \cos\left(\frac{-\beta \cdot \pi}{180}\right) \cdot xR3_1 - \sin\left(\frac{-\beta \cdot \pi}{180}\right) \cdot yR3_1 \quad (7.1)$$

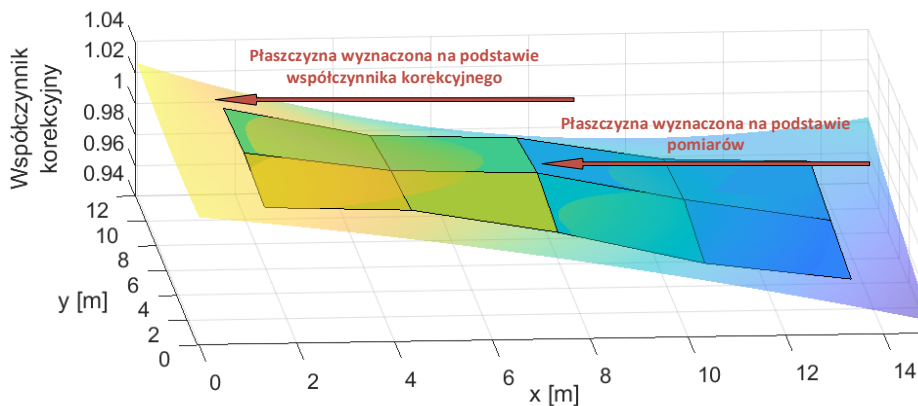
$$yR3_2 = \sin\left(\frac{-\beta \cdot \pi}{180}\right) \cdot xR3_1 + \cos\left(\frac{-\beta \cdot \pi}{180}\right) \cdot yR3_1 \quad (7.2)$$

$$\begin{aligned} \text{wsp}_k_{1m} = & 9,1812 \cdot 10^{-6} \cdot xR3_2^2 \cdot yR3_2^2 - 136,5 \cdot 10^{-6} \cdot xR3_2 \cdot yR3_2^2 + \\ & + 88,097 \cdot 10^{-6} \cdot yR3_2^2 - 58,190 \cdot 10^{-6} \cdot xR3_2^2 \cdot yR3_2 + \\ & + 1038,8 \cdot 10^{-6} \cdot xR3_2 \cdot yR3_2 - 778,2 \cdot 10^{-6} \cdot yR3_2 + \\ & - 115,1 \cdot 10^{-6} \cdot xR3_2^2 - 3300,6 \cdot 10^{-6} \cdot xR3_2 + 1,0227 \end{aligned} \quad (7.3)$$

$$\begin{aligned} \text{wsp}_k_{2m} = & 12,079 \cdot 10^{-6} \cdot xR3_2^2 \cdot yR3_2^2 - 237,3 \cdot 10^{-6} \cdot xR3_2 \cdot yR3_2^2 + \\ & + 815,6 \cdot 10^{-6} \cdot yR3_2^2 - 157,5 \cdot 10^{-6} \cdot xR3_2^2 \cdot yR3_2 + \\ & + 3361,8 \cdot 10^{-6} \cdot xR3_2 \cdot yR3_2 - 12828 \cdot 10^{-6} \cdot yR3_2 + \\ & + 583,6 \cdot 10^{-6} \cdot xR3_2^2 - 15362 \cdot 10^{-6} \cdot xR3_2 + 1,0615 \end{aligned} \quad (7.4)$$

gdzie:  $xR3_1, yR3_1$  – współrzędne względem układu  $R3_1$ ,  $xR3_2, yR3_2$  – współrzędne względem układu  $R3_2$ ,  $\beta$  – Kąt obrotu pomiędzy układem  $R3_1$ , a  $R3_2$   $\text{wsp}_k_{1m}, \text{wsp}_k_{2m}$  – współczynnik korekcyjny dla wysokości jednego i dwóch metrów.

Na rysunku 7.2 pokazano porównanie wartości współczynnika korekcyjnego wyznaczonego na podstawie funkcji aproksymującej dla całego obszaru  $LASL$  względem piętnastu punktów pomiarowych dla wysokości jednego metra w  $R3_1$ .



**Rys. 7.2.** Weryfikacja funkcji aproksymującej dla całego obszaru  $LASL$  względem piętnastu punktów pomiarowych dla wysokości jednego metra względem  $R3_1$

## 7.4 Hardware – panel obsługi

Podczas projektowania systemu kładziono duży nacisk na bezpieczeństwo jego użytkowania. Ze względu na charakter projektu (praca z urządzeniami latającymi) oraz fakt, że zaprojektowany system powstał od podstaw, wymagane było zaimplementowanie zabezpieczeń na wielu płaszczyznach pracy. Założono, że w razie wystąpienia braku łączności z panelem obsługi bądź wymaganymi UAV, awarią systemu nawigacji, wyjściem wskazań czujników poza skalę, rozładowaniem baterii, system zawsze ląduje.

Wszystkie te ewentualności mogą nastąpić na skutek zakłóceń, błędów programowych, uszkodzeń elektroniki bądź innych podzespołów. Przewidziano także awarie uniemożliwiające lądowanie. W takim momencie system przechodził w procedurę *failsafe* i wyłączał awaryjnie

wszystkie jednostki napędowe. Skutkowało to natychmiastowym sprowadzeniem urządzeń na ziemię (najczęściej z uszkodzeniami).

Panel obsługi (rys. 7.3) pozwalał na włączenie UAS oraz jego wyłączenie poprzez rozbrojenie silników. Jego obecność w sieci zawsze była obowiązkowa, a jego utrata wiązała się z natychmiastowym lądowaniem UAV. Planowanie i zaprogramowanie UAS wymaga wykorzystanie dedykowanego oprogramowania na komputerze klasy PC. Możliwy był jednak lot bez wizualizacji na komputerze.

Panel obsługi zawiera dwa grzybki bezpieczeństwa, włącznik główny, trzy kontrolki w kolorach czerwony, żółty i zielony oraz pięć diod sygnalizacyjnych, ponumerowanych odpowiednio 1, 2, 3, 4, 5. Informują one operatora, o jakim adresie urządzenia zostały wykryte przez system i odbierają dane. Na rysunku 7.3 pokazano przykład, w którym w sieć wpięte były dwa urządzenia o  $id = 1$  i 4. W prawej górnej części znajduje się antena do komunikacji z urządzeniami oraz port USB do komunikacji z komputerem klasy PC, na którym zainstalowano oprogramowanie wizualizacyjne pracę sieci.



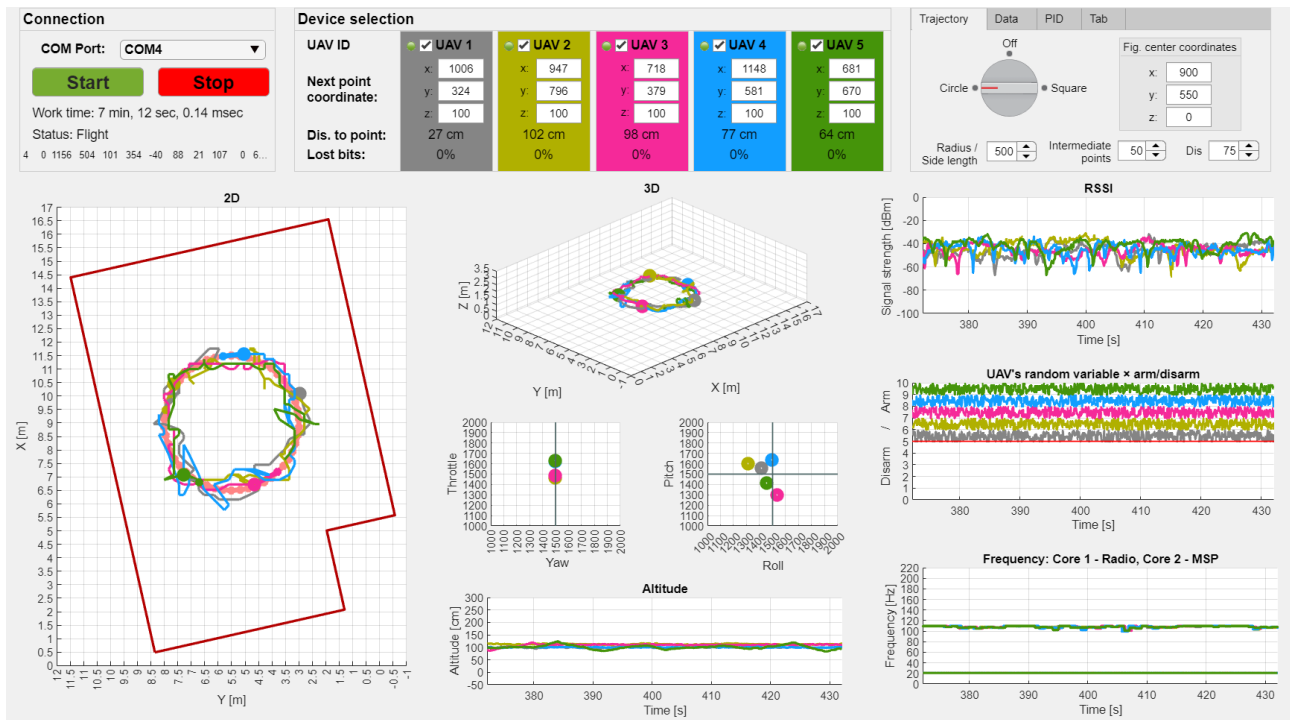
Rys. 7.3. Panel obsługi

## 7.5 Software – interfejs operatora

Wykorzystywany system pozwalał na podgląd pracy systemu w czasie rzeczywistym przy wykorzystaniu autorskiego interfejsu operatorskiego. W ramach pracy napisano jego dwie wersje z wykorzystaniem oprogramowania Matlab firmy MathWorks. Programy pozwalały na zapis danych w celu późniejszej weryfikacji.

Na rysunku 7.4 pokazano panel operatorski w wersji drugiej. Napisano go przy użyciu zaimplementowanego wewnątrz Matlab'a programu do tworzenia aplikacji graficznych App Designer. Głównym zadaniem panelu była wizualizacja przemieszczeń oraz stanu wybranych parametrów (RSSI, napięcie na baterii oraz sterowanie) wszystkich urządzeń znajdujących się w sieci.





Rys. 7.4. Panel operatorski v2.0

## 8. Algorytm dla przestrzeni otwartej

Decentralizacja pracy urządzeń wewnątrz sieci wymaga algorytmu wyznaczającego trajektorię dla każdego urządzenia z osobna, jednak w sposób od siebie zależny. Zakazane jest krzyżowanie się trajektorii w tej samej chwili, a także naruszenie stref bezpieczeństwa każdego urządzenia. Jako strefę bezpieczeństwa przyjmuje się obszar wokół każdego UAV określony niepewnością pomiarową stosowanego systemu nawigacji [X7].

W pracy poszukiwano rozwiązania, które gwarantowałoby wszystkim urządzeniom, nawet wykonującym obliczenia niezależnie, osiągnięcie tego samego rozwiązania. W niniejszej pracy pokazano algorytm generowania trajektorii dla grupy dronów w przestrzeni otwartej. Algorytm napisano w oparciu o następujące założenia:

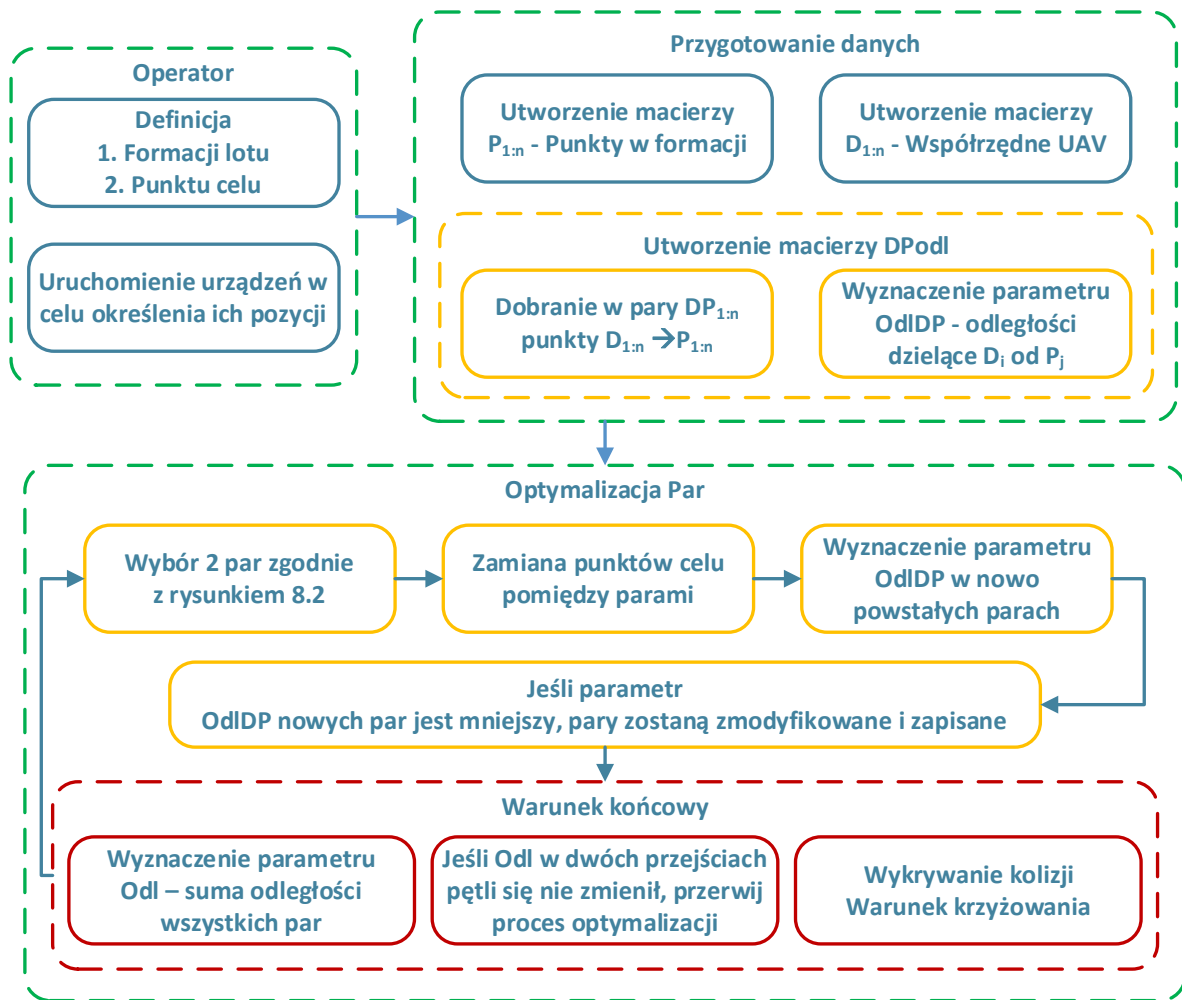
- Sterowanie systemu odbywa się na podstawie podania pośrednich punktów przelotowych (opcjonalnie), jednego punktu celu oraz formacji lotu.
- System sam decyduje o klasyfikacji pojedynczego urządzenia w grupie wyznaczając mu trajektorię wewnątrz grupy.
- Na podstawie pozycji każdego UAV, każde urządzenie jest w stanie niezależnie wyliczyć trajektorię każdego członka grupy.
- Powtarzalność wyniku – aby zapewnić decentralizację każde urządzenie (mimo niezależnie wykonywanych obliczeń) wylicza identyczną trajektorię dla każdego uczestnika grupy.
- Na podstawie pozycji i wyliczonej trajektorii system jest w stanie dokonać predykcji pozycji każdej jednostki w sieci.

### 8.1 Algorytm: Dobór w parę

#### 8.1.1. Klasyfikacja $D_{1:n} \rightarrow P_{1:n}$

Rozważając przestrzeń otwartą tzn. nie zawierającą żadnych granic ani przeszkód zaproponowano algorytm, który jest w stanie zarządzać nawet dużymi grupami urządzeń, spełniając

przy tym postawione na wstępie warunki. Rysunek 8.1. pokazuje schemat ideowy algorytmu. W dalszej części rozdziału opisano poszczególne jego bloki.



Rys. 8.1. Schemat ideowy algorytmu

### Pierwszy krok – Generowanie par

Generowanie par  $D_{1:n} \rightarrow P_{1:n}$  według początkowego adresu urządzenia i punktu. Jest to kryterium wymagane, aby każde urządzenie wykonując obliczenia równoległe, otrzymało taki sam wynik. Ideę doboru pokazano na rysunku 8.2. Para  $x$  prezentuje aktualną parę (np.  $D_1 \rightarrow P_1$ ,  $D_2 \rightarrow P_2$  itd.).

	$P_1$	$P_2$	$P_3$	$P_n$
$D_1$	Para 1: $P_1D_1$			
$D_2$		Para 2: $D_2P_2$		
$D_3$			Para 3: $D_3P_3$	
$D_n$				Para 4: $D_nP_n$

Rys. 8.2. Początkowa klasyfikacja  $D_{1:n} \rightarrow P_{1:n}$

### Drugi krok – Zmiany w parach

Na rysunku 8.3. pokazano ideę zamiany pomiędzy parami. Algorytm wybiera dwie pary (jedna zielona komórka) i liczy dystans ( $Dis_{DP}$ ) dla dwóch par w wersji aktualnej i zamienionej (zmiana punktów celu). Jeśli po zamianie suma odległości pomiędzy nowo uzyskanymi parami jest mniejsza niż przed zmianą, zostaje ona na stałe. Np. dla komórki „P21” program porówna sumę  $Disp_{D1P1}$ ,

$Disp_{D2P2}$  z sumą  $Disp_{D1P2}$ ,  $Disp_{D2P1}$ . Takie działanie zapobiega krzyżowaniu się drogi pomiędzy parami i stałemu zmniejszeniu odległości dla całej sieci.

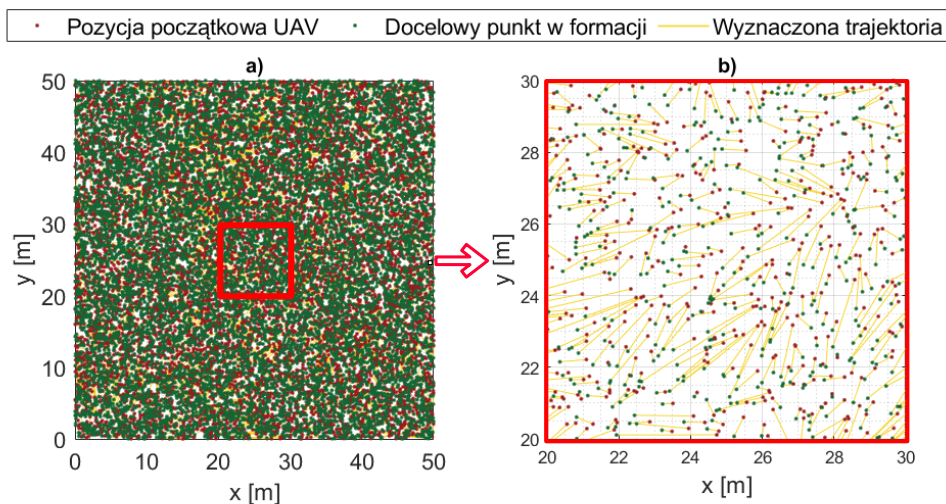
	Para 1	Para 2	Para 3	Para n
Para 1				
Para 2	Para 2 – Para 1			
Para 3	Para 3 – Para 1	Para 3 – Para 2		
Para n	Para n – Para 1	Para n – Para 2	Para n – Para 3	

Rys. 8.3. Porównanie par – schemat ideowy

### Trzeci krok – Warunek końcowy

Jeśli algorytm przejdzie wszystkie kroki (w tabeli na rysunku 8.3), to wyznaczany jest parametr  $Dis$  – suma pomiędzy wszystkimi parami  $D_{1:n} \rightarrow P_{1:n}$  w grupie. Jeśli parametr  $Dis$  w dwóch przejściach pętli jest taki sam, program zapisuje uzyskane pary i przerywa poszukiwania minimum. Nie jest możliwe dalsze zmniejszenie parametru  $Dis$ .

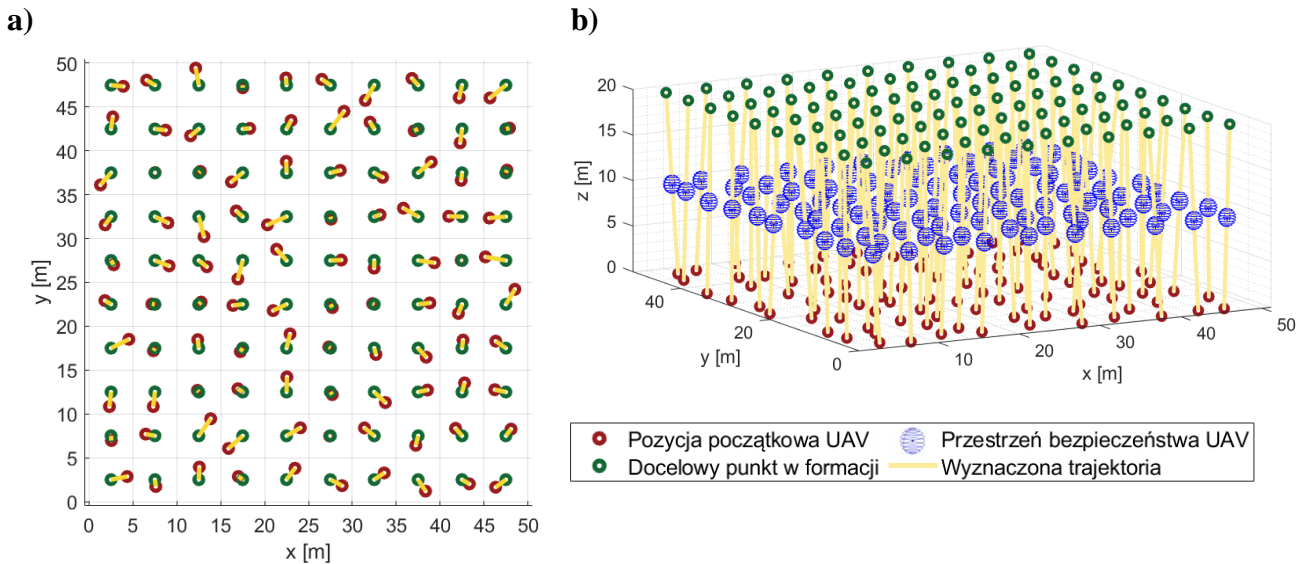
Warunkiem sprawdzającym dla całego systemu był test krzyżowania się dróg cząstkowych. Idea algorytmu eliminuje dopuszczenie takiej możliwości, gdyż geometrycznie zawsze suma dwóch krzyżujących się tras jest większa od ich alternatywnego połączenia. Na rysunku 8.4a pokazano symulację dla 10 000 tras. Pozycję urządzeń oraz ich punkty docelowe wygenerowano losowo. Rysunek 8.4b pokazuje przybliżenie pewnego kawałka przestrzeni. Pokazany przypadek w rzeczywistych warunkach jest nierealny ponieważ pozwala na generowanie punktów bez zachowania odpowiedniego dystansu – miał on jedynie na celu pokazanie, iż nawet dla takiego przykładu, żadne drogi się nie krzyżują. Przedstawiony przykład wskazuje na potrzebę poprawnego umiejscowienia punktów startu i celu. Problemem były położone zbyt blisko siebie równoległe trasy. Pomimo, iż warunek krzyżowania spełniono, fizycznie urządzenia mogą się dotknąć.



Rys. 8.4. Symulacja dla 10 000 urządzeń a) przestrzeń całkowita, b) przybliżenie

## 8.2 Symulacja lotu

Pierwsza symulacja zakładała lot pionowy 100 UAV w górę do formacji kwadratowej. Rysunek 8.5a pokazuje przydział (2D)  $D_{1:n} \rightarrow P_{1:n}$ . Analiza wykresu wskazuje, że prawdopodobnie nie można połączyć urządzenia z punktami celu uzyskując mniejszy sumaryczny dystans dla całego systemu [X16]. Program już w drugiej iteracji znalazł najoptymalniejsze rozwiązanie, a czas obliczeń wyniósł 16 ms. Rysunek 8.5b przedstawia symulację w 3D. Niebieskie kule określają zadeklarowaną przestrzeń bezpieczeństwa każdego urządzenia.



Rys. 8.5. Trajektoria UAV do formacji kwadratowej: a) 2D, b) 3D

## 9. Algorytm dla przestrzeni zamkniętej

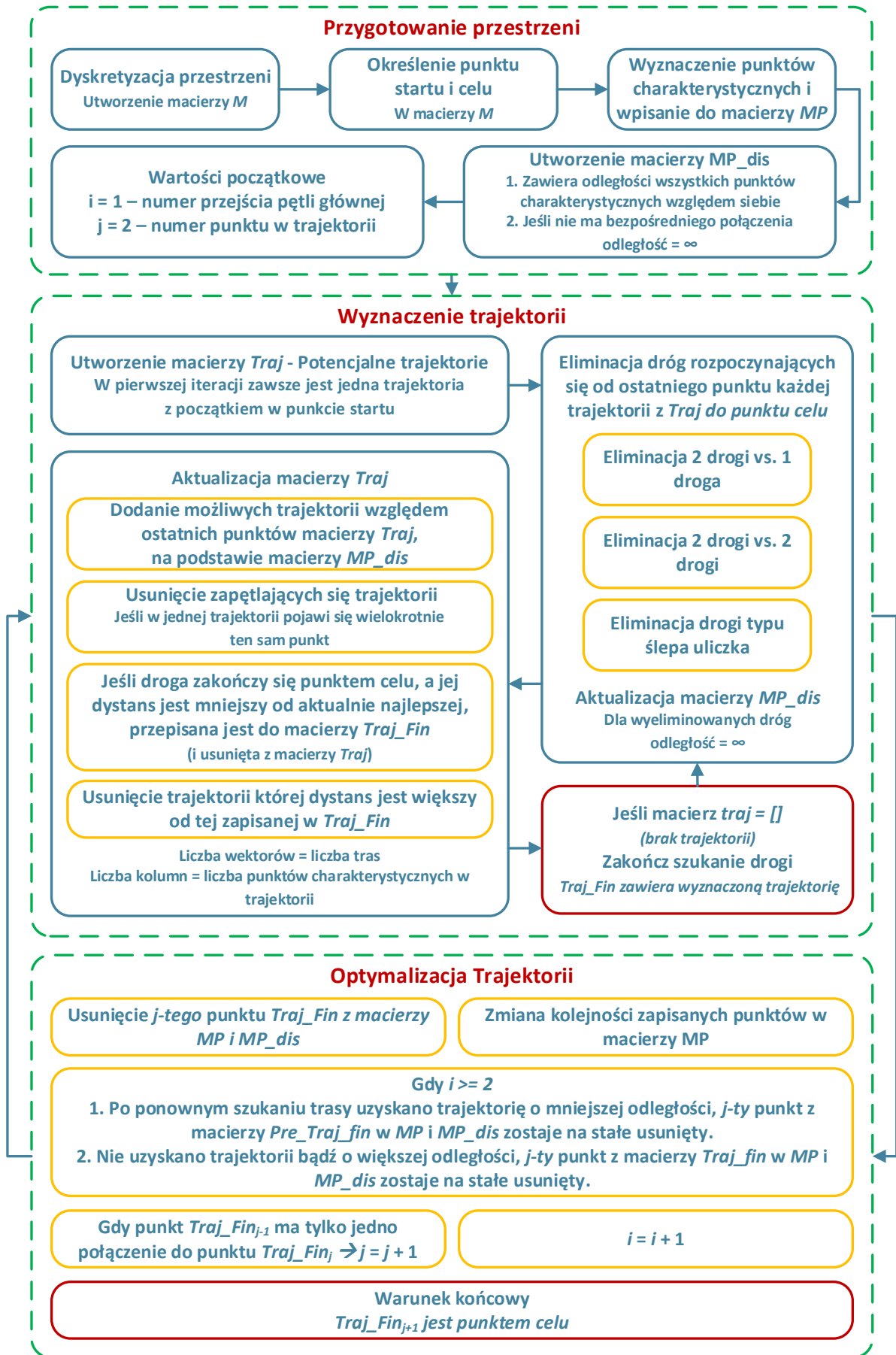
W rozdziale ósmym omówiono algorytm zarządzania grupą urządzeń w przestrzeni otwartej. Rozumie się przez to, że przestrzeń nie posiada granic, a wewnątrz niej nie mogły znajdować się żadne przeszkody. Założenie to nie jest błędne, ponieważ w pewnych zastosowaniach można założyć takie warunki, np. podczas pokazów świetlnych. Najczęściej jednak urządzenia pracują w przestrzeni, w której mogą znajdować się jakieś przeszkody, a UAS powinno aktualizować swoją trajektorię w celu zapewnieniu bezpieczeństwa urządzeń [7]. W niniejszej pracy przestrzeń taką nazwano zamkniętą bądź labiryntem. Celem algorytmu jest znalezienie możliwie najkrótszej drogi, łączącej dwa punkty, startu oraz celu, przy ominięciu przeszkód znajdujących się na drodze drona. Znalezienie optymalnej drogi zmniejsza czas oraz energię potrzebną na dotarcie do punktu końcowego [6]. W tym rozdziale zostanie pokazane rozwiązanie pozwalające na wyznaczenie drogi metodą analityczną zgodnie z założeniami:

- Program powinien znaleźć quasi-optymalną drogę wyjścia z labiryntu.
- Powtarzalność wyniku, tzn. przy wielokrotnym wyznaczeniu trasy o tych samych warunkach początkowych wynik powinien być identyczny.
- Granice i przeszkody wewnątrz całego labiryntu są znane.

Analiza przeglądu literaturowego wykazała, że najczęściej stosowane rozwiązania oparte są na algorytmach genetycznych [18, 24, 25]. Stochastyczne podejście nie gwarantuje jednak znalezienia najkrótszej drogi oraz powtarzalności wyniku.

### 9.1 Algorytm: rozwiązanie labiryntu metodą analityczną

Napisany algorytm wymagał, aby przestrzeń oraz znajdujące się wewnątrz niej przeszkody zostały podane w formie współrzędnych punktów leżących na jego wierzchołkach. Na rysunku 9.1 pokazano schemat ideowy proponowanego algorytmu.



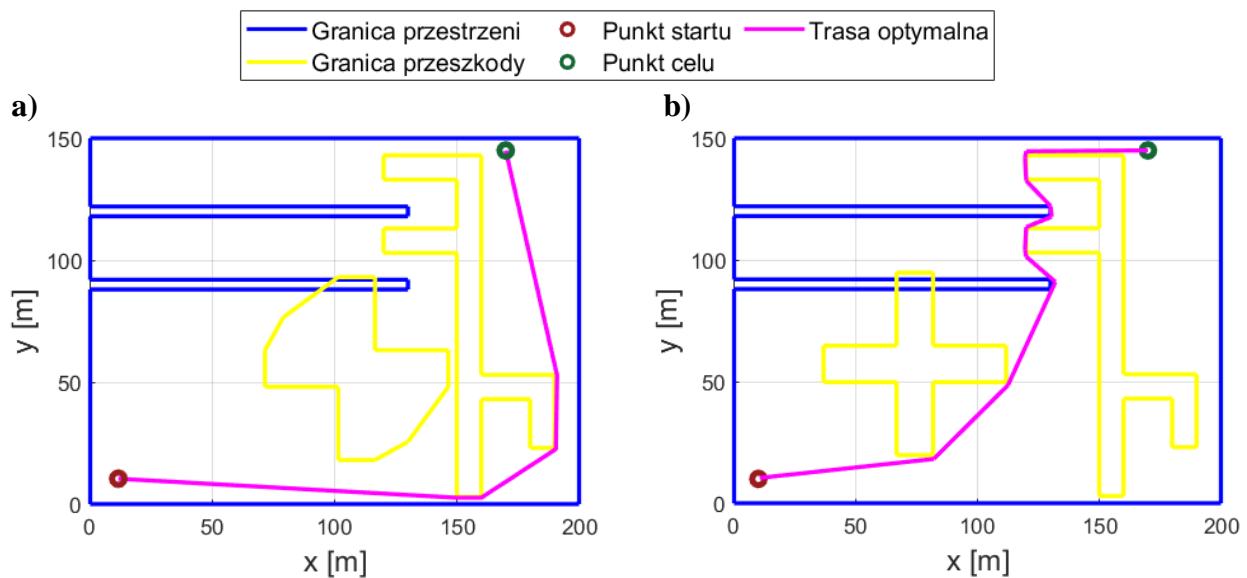
Rys. 9.1. Schemat ideowy pracy algorytmu



Agenda do schematu ideowego

- $M$  – macierz opisu przestrzeni
- $MP$  – macierz zawierająca współrzędne punktów charakterystycznych
- $MP_{dis}$  – macierz zawierająca odległości każdego punktu charakterystycznego względem każdego innego punktu charakterystycznego
- $i$  – numer przejścia pętli wewnętrznej, eliminacji dróg
- $j$  – numer kolejnego punktu charakterystycznego wewnątrz wyznaczonej trajektorii
- $Traj$  – robocze trajektorie
- $Traj_{fin}$  – Trajektoria z przejścia  $i$
- $Pre\_Traj_{fin}$  – Trajektoria z przejścia  $i-1$

Na rysunku 9.2 a,b przedstawiono rozwiązanie kilku przykładowych labiryntów. Każdorazowo algorytm znalazł najkrótszą drogę wyjścia z labiryntu.



Rys. 9.2. Wyznaczona trajektoria dla przykładowych labiryntów z wykorzystaniem proponowanego algorytmu

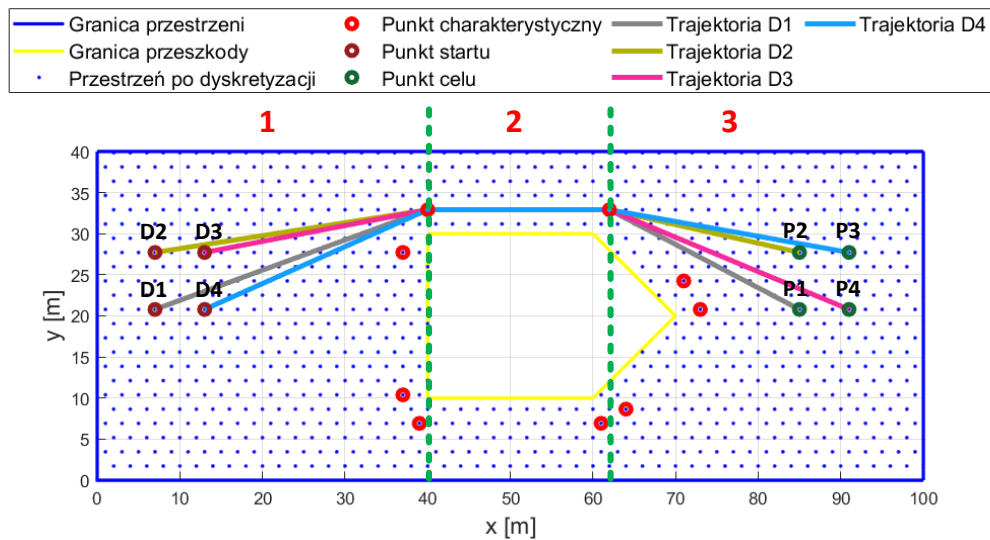
## 9.2 Symulacja lotu

Implementację algorytmu dla jednego UAV wykonano w bezpośredni sposób, tzn. przyjęto, że znaleziona trasa jest trajektorią urządzenia. Dla bezzałogowego systemu latającego wymagane było przyjęcie warunku stałej bądź zmiennej formacji, tzn. ustalenie, czy urządzenia w trakcie lotu mogą (bądź nie) zmieniać pozycję względem siebie. Przyjęcie stałej formacji nie zawsze jest możliwe, np. przy wąskich przesmykach labiryntu, stąd podczas badań dopuszczano jej zmianę. Ponadto utrzymywanie stałej formacji jest nieoptymalne choćby podczas omijania prostych przeszkód, np. urządzenia będące po zewnętrznej stronie łuku pokonują dłuższą drogę.

W ramach symulacji przyjęto dwa warianty sterowania. Pierwszy – niezależny, zakładał znajdowanie dla każdej pary  $D \rightarrow P$  oddzielnej trajektorii. Rozwiązanie to było jednak problematyczne podczas wyszukiwania konfliktów trajektorii w celu uniknięcia kolizji w UAS.

Drugi wariant pokazano na rysunku 9.3. Przykład przedstawia wygenerowaną trajektorię dla UAS przy założeniu, że formacja może się zmieniać w trakcie lotu, a trajektorie są względem siebie zależne. Trajektorii poszukiwano dla średniej pozycji urządzeń i punktów celu. Każde urządzenie w pierwszym sektorze znajduje drogę do pierwszego punktu przelotowego. Unikanie zderzeń UAV wykonano zakładając różną prędkość przelotową na pierwszym etapie lotu. Kontrola kolizji nie dopuszczała do sytuacji, w której w macierzy  $M(:, :, 5)$  jakkolwiek wartość była większa od 1, dając

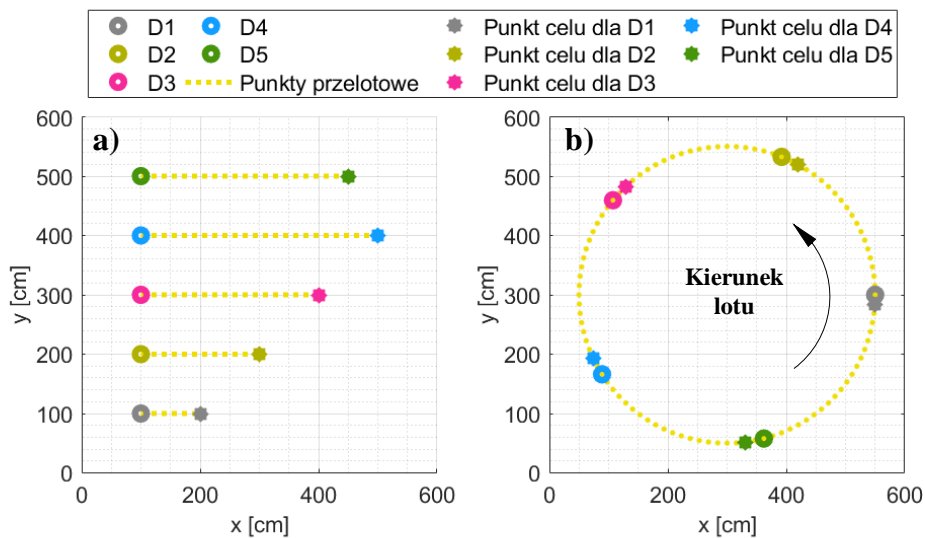
pierwszeństwo urządzeniu będącemu najbliżej kolejnego punktu przelotowego trajektorii. W drugim sektorze urządzenia poruszają się jeden po drugim. Ostatni (trzeci) sektor pozwala na lot w kierunku docelowego punktu w formacji.



Rys. 9.3. Wyznaczona trajektoria dla UAS – podział na sektory

## 10. Badania na obiektach rzeczywistych

W niniejszym rozdziale zamieszczono uzyskane wyniki badań z wykorzystaniem minimum dwóch urządzeń. Zgodnie z założeniami postawionymi w rozdziale drugim, przeprowadzono kontrolowane loty dla sieci o maksymalnym rozmiarze pięciu UAV. Zarządzanie siecią oraz sterowanie UAV odbywało się z zachowaniem wszystkich warunków decentralizacji.



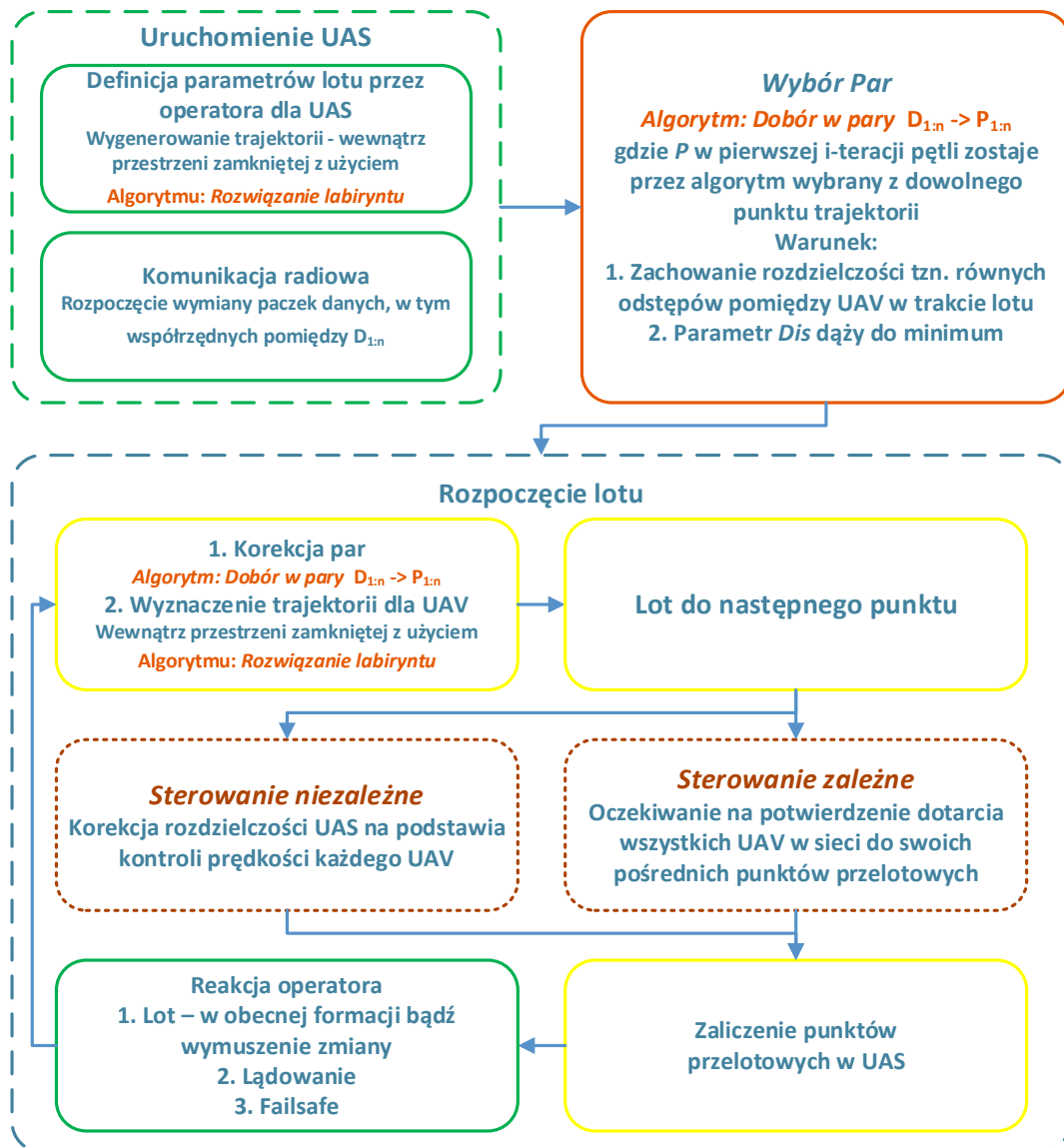
Rys. 10.1. a) Sterowanie niezależne, b) Sterowanie zależne

Rozdział podzielono na dwa podrozdziały. W pierwszym uwzględniono sterowanie niezależne tzn. każde urządzenie wyznacza swoją, niezależną trajektorię nieleżącą na linii innego urządzenia (rys. 10.1a). Zadaniem sieci (w rozumieniu ogólnym) jest kontrolowanie lotu oraz zapewnienie braku konfliktów (kolizji), jednak z samego założenia, UAV mogą poruszać się wewnątrz swojej trajektorii

swobodnie. Drugi podrozdział uwzględnia sterowanie zależne (rys. 10.1b). Rozumie się przez to zarządzanie trajektorią poszczególnych urządzeń wewnątrz sieci w sposób zależny, tzn. urządzenie, aby mogło poruszać się po swojej trajektorii, musi kontrolować trajektorię innych UAV, gdyż znajdują się one na jego bezpośrednim torze lotu.

Rysunek 10.2 przedstawia schemat ideowy pracy sieci. Bezpośrednio po uruchomieniu panelu operatorskiego system wykrywa i łączy się z UAV. W zależności od liczby połączonych urządzeń system autonomicznie zarządza siecią. Rolą operatora jest wybór zestawień kształtów, wzdłuż których urządzenia poruszają się oraz podniesienie lewego grzybka bezpieczeństwa (na panelu obsługi) w celu fizycznego potwierdzenia startu. Po wyborze kształtów system pierwotnie sprawdzał czy na przestrzeni lotu znajdują się jakieś ograniczenia (przedmioty). Wykrycie przeszkód skutkowało użyciem algorytmu *Rozwiązanie labiryntu* (opisanego w rozdziale 9.1) w celu znalezienia alternatywnej drogi. W przypadku sterowania zależnego punkty celu szukano z docelowego kształtu. Algorytm *Dobór w parę*, z uwzględnieniem zachowania rozdzielczości, dokonywał wyboru punktów dla których suma przebytych dróg była najkrótsza dla całego UAS.

Po wyznaczeniu punktu celu dla każdego UAV system jest gotowy do lotu. Operator potwierdza start podnosząc lewy grzybek panelu obsługi (przy założeniu, że prawy również był podniesiony).



Rys. 10.2. Schemat ideowy

W trakcie badań zauważono, iż przy odległościach kilkudziesięciu centymetrów dochodziło do zakłamania pozycji przez system nawigacji. Urządzenie nawet poruszając się po linii prostej, zwracało koordynaty w kształcie zygzaka. Miało to także wpływ na sterowanie wewnątrz UAS, gdyż pojedynczy UAV nie potrafi się nagle zatrzymać (inercja układu). Warunek zaliczenia punktu ustawiono w zakresie 50-75 cm. Podczas lotu pięciu urządzeń wszystkie pięć UAV powinny znajdować się w takiej odległości od swojego punktu celu, aby system mógł uznać go za zaliczony.

## 10.1 Sterowanie zależne

Jako trajektorię zależną rozumie się wyznaczenie trajektorii w taki sposób, aby urządzenia mogły poruszać się jeden po drugim. Dodatkowo początek i koniec może być ze sobą połączony, a UAV mogą poruszać się w niej zapętłając w figurze. W ramach badań wybrano okrąg oraz kwadrat o różnych wymiarach. Urządzenia poruszały się do momentu komendy od operatora wymuszającej lądowanie.

### 10.1.1. UAS $n = 5$ – wyniki szczegółowe

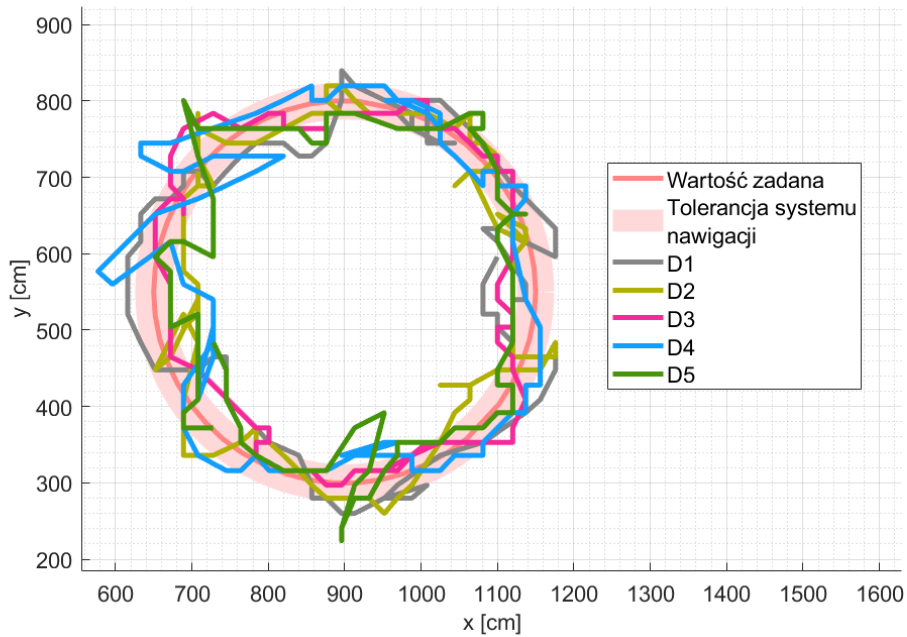
Podstawowym celem niniejszej pracy było wykonanie kontrolowanych lotów UAS, w której skład wchodzi pięć UAV. Rysunek 10.3 przedstawia zdjęcie UAS w trakcie lotu pięciu urządzeń po okręgu o średnicy 600 cm. W dalszej części pokazano wyniki szczegółowe z doświadczenia, podczas którego zrobiono zdjęcie.

Na wykresach przedstawiono najistotniejsze informacje rejestrowane przez system. Opisując UAS najważniejszym parametrem jest jego przemieszczenie. Na rysunku 10.4 pokazano zarejestrowaną pozycję przy trajektorii w kształcie okręgu o średnicy pięciu metrów. Urządzenia tuż po starcie i osiągnięciu wysokości jednego metra poruszały się zgodnie z ruchem wskazówek zegara, utrzymując równy dystans pomiędzy sobą. Na wykresie zaznaczono płaszczyznę tolerancji systemu nawigacji. Maksymalna rozbieżność od trajektorii idealnej wynosi 50 cm.

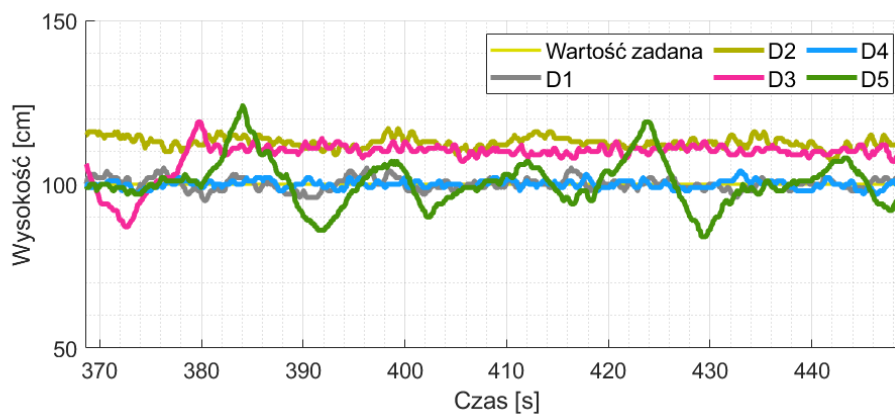
Urządzenia w trakcie lotu miały utrzymywać wysokość 100 cm. Zauważono, przy lotach z większą liczbą urządzeń, iż występują oscylacje wysokości (rys. 10.5). System sterowania uwzględnia także wskazania z nawigacji Marvelmind oraz barometrycznego czujnika wysokości. Okazuje się, że tego typu oscylacje występują przy 4 – 5 UAV, co wiąże się z ograniczeniami systemu LPS.



*Rys. 10.3. Pięć urządzeń w trakcie lotu po okręgu o średnicy 600 cm*



**Rys. 10.4.** Pomiar pozycji pięciu UAV w trakcie lotu po okręgu

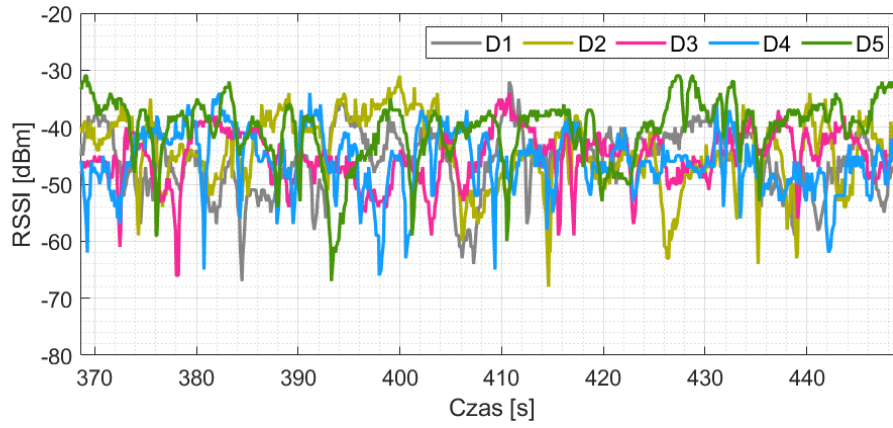


**Rys. 10.5.** Wskazanie LIDARU pięciu UAV w trakcie lotu po okręgu

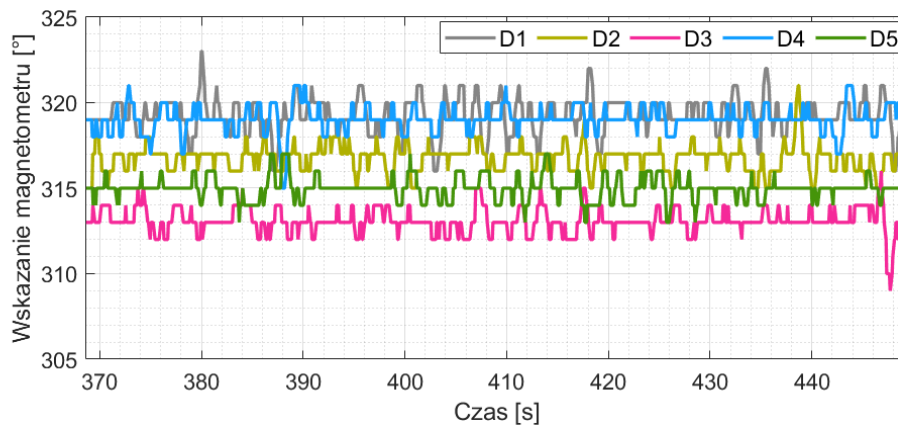
W ramach pracy wiele czasu poświęcono komunikacji radiowej. Na rysunku 10.6 pokazano zarejestrowany na każdym urządzeniu parametr RSSI (UAV → panel obsługi). W tym miejscu należy nadmienić, iż antenę panelu obsługi wyciągnięto poza klatkę bezpieczeństwa operatora i powieszono na wysokości dwóch metrów na ścianie laboratorium. Działanie to miało wyeliminować wpływ urządzeń, (komputera, monitora, ładowarek baterii, lamp, aparatury warsztatowej) na jakość transmisji. Antenę połączono z panelem z wykorzystaniem dedykowanego kabla koncentrycznego. Analiza wyników wskazuje, iż w trakcie całego lotu, wszystkie urządzenia działały stabilnie. Podczas pomiarów nie dochodziło do nieplanowanych rozłączeń bądź usunięć z sieci.

Wskazania magnetometru (rys. 10.7) potwierdziły poprawność działania funkcji aproksymującej, korygującej jego wskazanie. Urządzenia utrzymywały w trakcie całego lotu, zapisany przy starcie kierunek. Rozbieżność w trakcie lotu wynosiła około  $3^\circ$ . System nie wymagał dokładnego ustawienia urządzeń w początkowej fazie na wartości  $315^\circ$ . Niewielkie rozbieżności nie wpływały znacząco na jakość lotu.

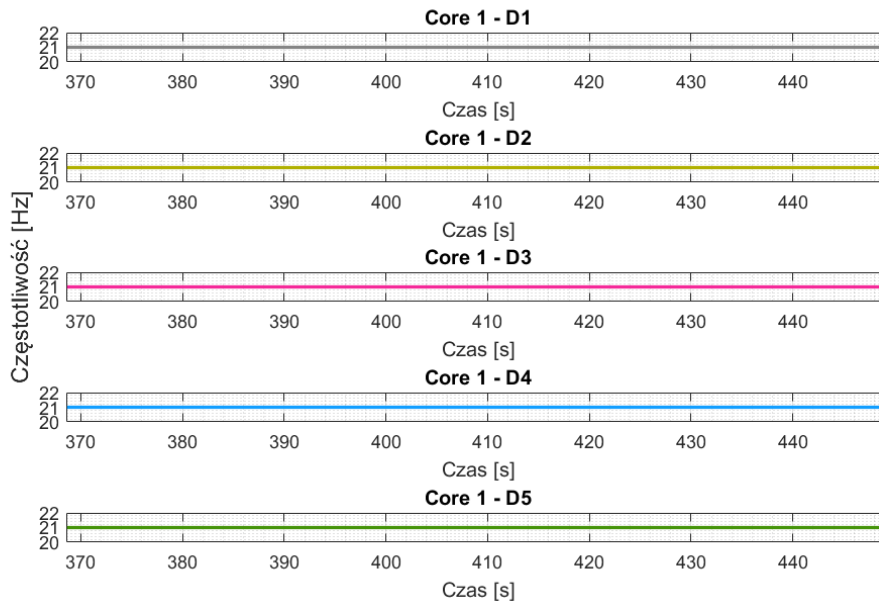




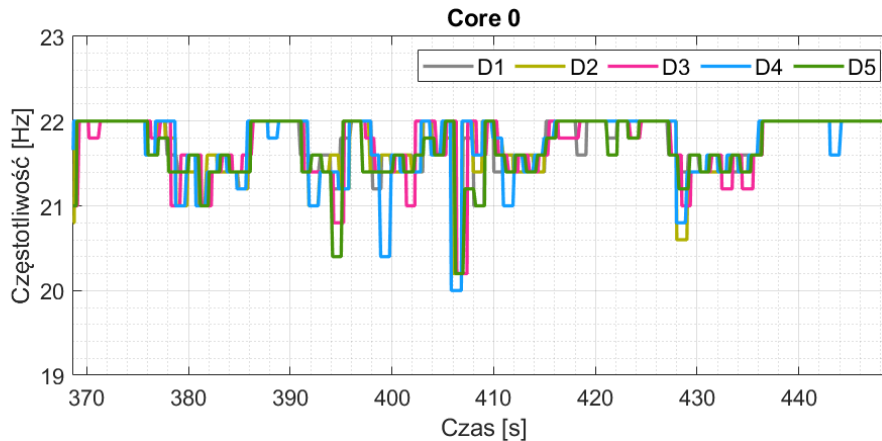
Rys. 10.6. Parametr RSSI dla pięciu UAV w trakcie lotu po okręgu



Rys. 10.7. Wskazanie magnetometru pięciu UAV w trakcie lotu po okręgu



Rys. 10.8. Częstotliwość aktualizacji pętli sterowania

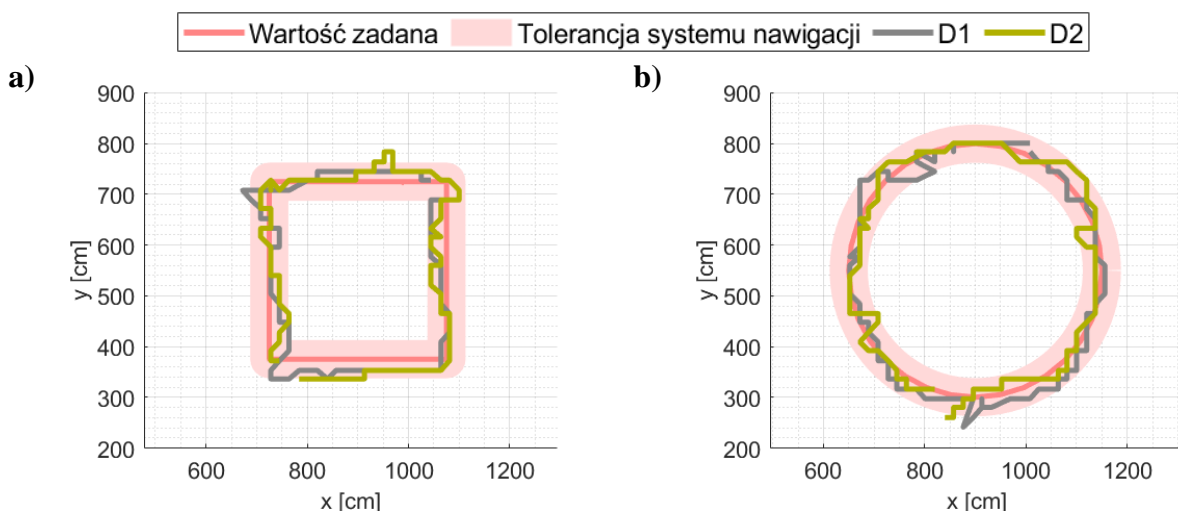


**Rys. 10.9.** Częstotliwość wymiany paczek danych pomiędzy każdym UAV

Na rysunkach 10.8 i 10.9 zaprezentowano częstotliwość pracy dwóch wątków procesora ESP - 32. Pierwszy rysunek pokazuje częstotliwość wątku drugiego, w ramach którego pobierane są wszystkie informacje z kontrolera lotu przez ESP - 32 i na ich podstawie wyznaczone parametry sterowania. Częstotliwość pracy wątku była stabilna na poziomie 21 Hz dla wszystkich UAV. Na rysunku 10.9 pokazano częstotliwość z jaką dokonywana jest wymiana paczek danych pomiędzy urządzeniami - każde urządzenie aktualizuje pozycję każdego innego urządzenia z częstotliwością 20 – 22 Hz. Rozbieżność 2 Hz wynika z faktu, iż czas transmisji uzależniony jest od odległości, ustawiania anteny oraz parametru RSSI. Parametry te nie są stałe.

### 10.1.2. UAS $n = 2$

Na rysunkach 10.10 pokazano zarejestrowane przemieszczenie dwóch urządzeń, wykonujących lot po okręgu (10.10b) i kwadracie (10.10a). Na uwagę zasługuje wysoka dokładność UAS mieszcząca się w wyznaczonym torze ruchu urządzeń, wynikającym z tolerancji systemu nawigacji. Możliwy był także lot po stosunkowo małych figurach tzn. okręgu i kwadracie nieprzekraczający 250 cm (10.10a).

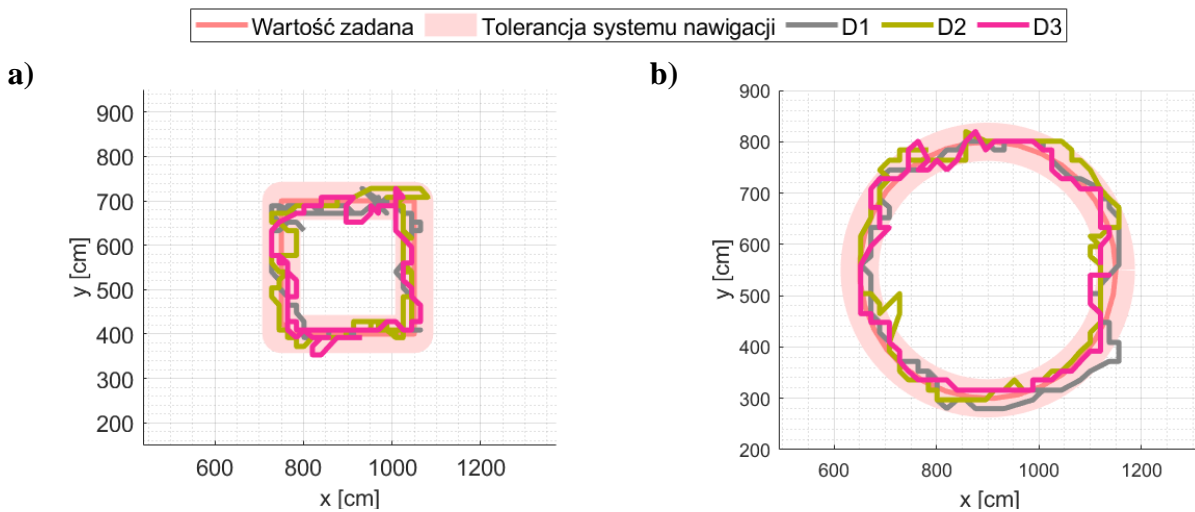


**Rys. 10.10.** Wybrane formacje lotu **a)** kwadrat o boku 250 cm, **b)** okrąg o średnicy 500 cm

### 10.1.3. UAS $n = 3$

Na rysunkach 10.11 pokazano zarejestrowane przemieszczenie, gdy UAS zawierał trzy urządzenia. Trajektorię wygenerowano ponownie wykorzystując kształt kwadratu (10.11a) i okręgu

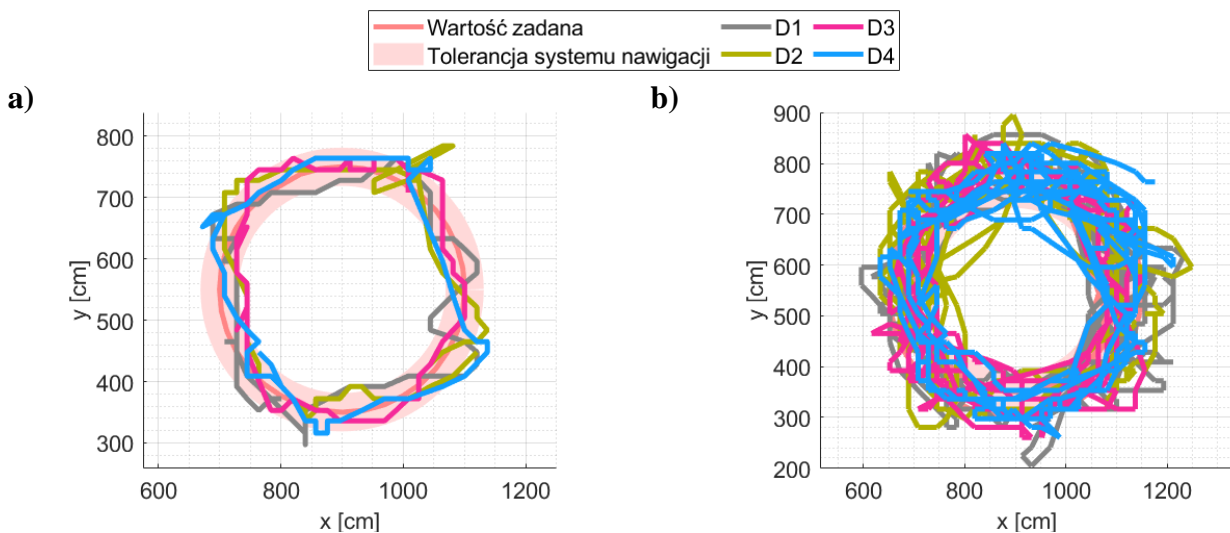
(10.11b). Na rysunku 10.11a pokazano pokonaną drogę urządzeń poruszających się po kwadracie o boku 200 cm. Ruch po małej figurze był trudny do osiągnięcia ponieważ w okolicy naroży urządzenia musiały wyhamowywać swoją prędkość (wzdłuż obecnej osi ruchu) do zera. W początkowych etapach widoczne było także przeregulowane urządzeń, tzn. wychodzenie poza krawędź kwadratu.



**Rys. 10.11.** Wybrane formacje lotu **a)** kwadrat o boku 250 cm, **b)** okrąg o średnicy 500 cm

#### 10.1.4. UAS $n = 4$

Rysunek 10.12a przedstawia zarejestrowaną trajektorię lotu czterech urządzeń po okręgu o średnicy 400 cm. Na rysunku 10.12b pokazano pełną trajektorię, od startu do lądowania. Pełny tor lotu jest szerokości około 100 cm. Spowodowane jest to błędami w stosowanym systemie nawigacji, który przy rosnącej liczbie użytkowników, zmniejszał częstotliwość odświeżania pozycji oraz był bardziej podatny na zakłócenia.



**Rys. 10.12.** Wybrane formacje lotu **a)** okrąg o boku 400 cm, **b)** przebyta droga od startu przez 6 okrężeń do lądowania

## 10.2 Sterowanie niezależne

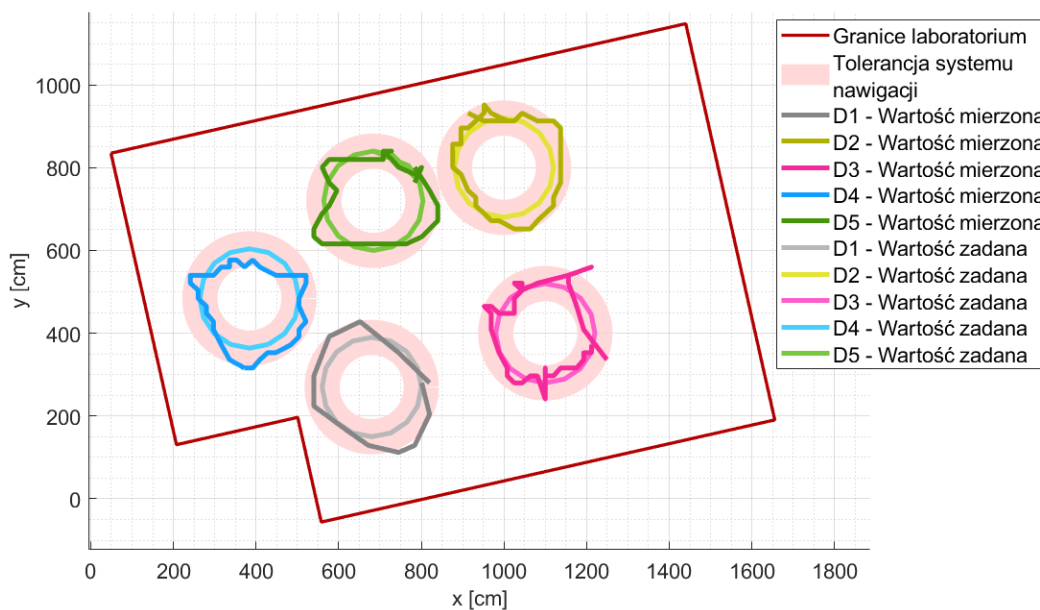
Pokazane w rozdziale 10.1 wyniki wskazują jednoznacznie, iż główne założenie pracy tzn. lot z wykorzystaniem pięciu bezzałogowych urządzeń latających jest możliwy. Niestety, zarejestrowana

droga jest poszarpana, wychodząc nawet poza deklarowaną tolerancję systemu nawigacji. Wynika to z faktu, iż trajektoria lotu każdego urządzenia jest zależna od trajektorii pozostałych urządzeń, tzn. aby urządzenie mogło zaliczyć punkt przelotowy musi oczekiwać, aż wszystkie inne również to zrobią. W taki sposób uzyskiwano pewność, że nie dojdzie do kolizji oraz można było zsynchronizować pewne ruchy. Niestety, błędy systemu pozycjonowania doprowadziły do lekkich opóźnień w dotarciu wybranych UAV do celu, wymuszając zatrzymanie innych. W tym rozdziale pokazano uzyskane wyniki przy założeniu, że każde urządzenie posiada swoją własną niezależną trajektorię lotu. Każde urządzenie po zaliczeniu swojego punktu przelotowego, mogło lecieć do następnego. Celem tych badań było sprawdzenie wpływu zależności/niezależności trajektorii na uzyskane wyniki.

### 10.2.1. UAS n = 5

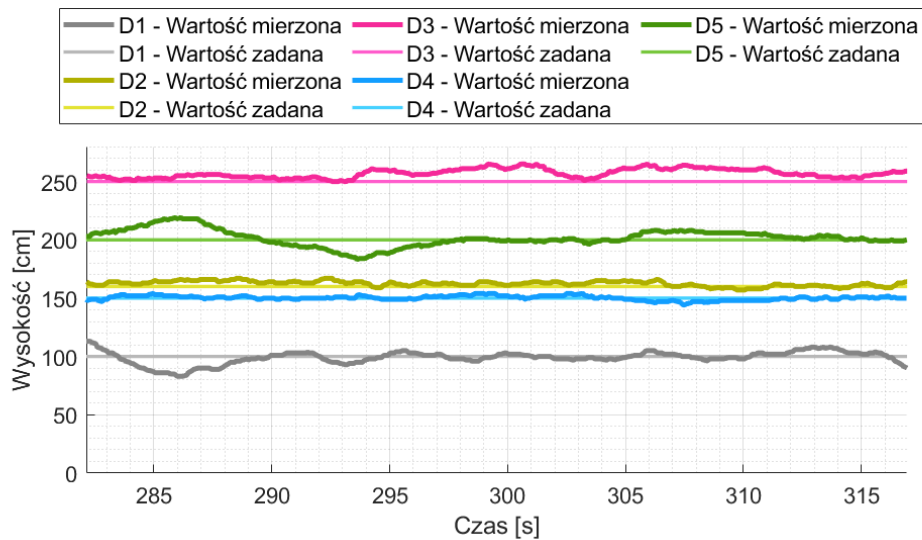
Rysunek 10.13 przedstawia wyniki uzyskane podczas lotów z wykorzystaniem pięciu urządzeń. Podczas badań urządzenia wykonywały ruch po okręgu o średnicy 200 cm. Pokazana na rysunku trajektoria uwzględnia tylko jeden obrót. Analiza wyników wskazuje, iż urządzenia także miały problem z idealnym pokonaniem trasy co szczególnie dobrze widać na przykładzie urządzenia D3, którego wskazania często wychodziły poza deklarowaną tolerancję.

Wykonanie tych badań było trudne. Pomimo iż trajektorie nie zachodziły na siebie, a symulacja wykazała, że urządzenia będą minimum trzy metry od siebie, dochodziło do ich kolizji. Spowodowane było to nierównym (czasowo) pokonaniem deklarowanej trasy. Dodatkowo nakładające się błędy systemu pozycjonowania nie pozwalały aby droga UAV idealnie odwzorowywała zadaną trajektorię.



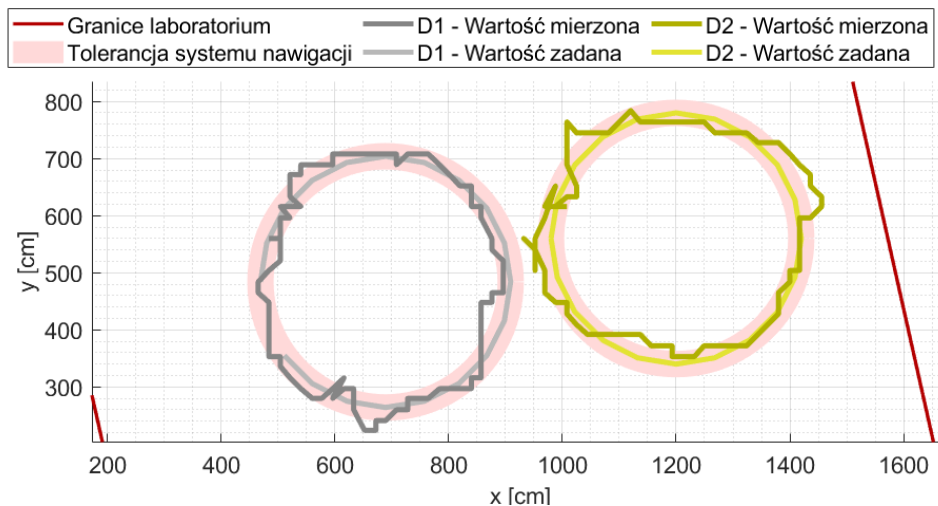
**Rys. 10.13.** Sterowanie niezależne – droga pięciu urządzeń wzdłuż trajektorii okręgu o promieniu dwóch metrów

Na rysunku 10.14 pokazano zmierzoną wysokość urządzeń w trakcie lotu pokazanego na rysunku 10.13. Celowo wybrano różne wysokości dla każdego z urządzeń, aby w razie błędów systemu nawigacji uniknąć kolizji. Na podstawie wyników można stwierdzić, że urządzenia dobrze utrzymywały deklarowaną wysokość.



Rys. 10.14. Sterowanie niezależne – wysokość

### 10.2.2. UAS n = 2



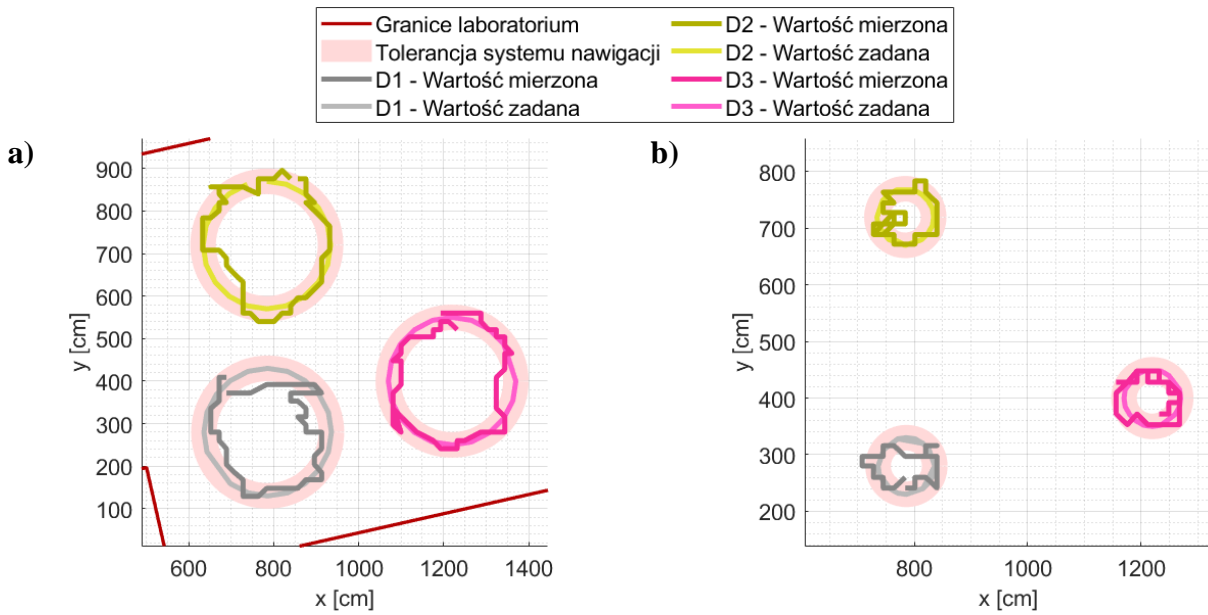
Rys. 10.15. Sterowanie niezależne dwóch UAV – okrąg o średnicy 400 cm

Na rysunku 10.15 pokazano lot niezależny dwóch urządzeń po okręgu o średnicy 400 cm. Analiza rysunku wskazuje iż urządzenia nie pokonują synchronicznie pełnego okręgu: D<sub>1</sub> nie zamknął jeszcze okręgu (dolna część wykresu), za to D<sub>2</sub> rozpoczął już nowy okrąg. Taka sytuacja nie wystąpiłaby w sterowaniu zależnym.

### 10.2.3. UAS n = 3

Na rysunkach 10.16 pokazano zarejestrowaną drogę UAS przy wykorzystaniu trzech urządzeń. Wybrano dwie trajektorie o kształcie okręgów i średnicach 300 cm (10.16a) i 100 cm (10.16b). Ze względu na ograniczone wymiary laboratorium nie można było zdefiniować większych trajektorii. Analiza wykresów wskazuje iż, pomimo niskiej częstotliwości odświeżenia pozycji przez LPS (2 – 4 Hz) system był w stanie wykonać kontrolowany lot.

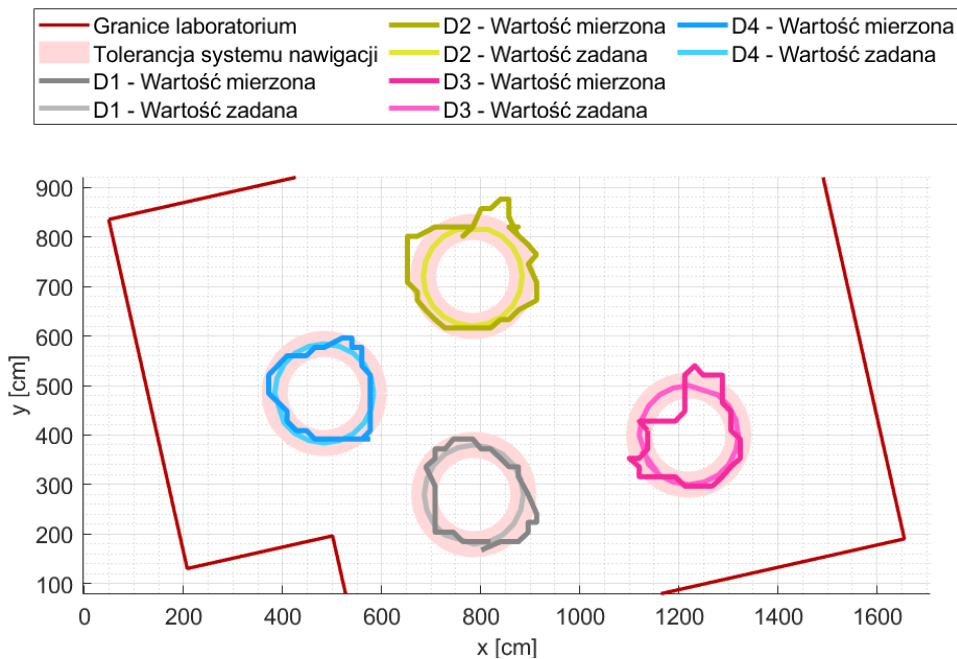




Rys. 10.16. Sterowanie niezależne trzech UAV – przemieszczenie  $x, y$  (2D), a) okrąg o średnicy 300 cm, b) okrąg o średnicy 100 cm

#### 10.2.4. UAS $n = 4$

Ostatni przykład przedstawia lot z wykorzystaniem czterech UAV wewnątrz UAS. Każde urządzenie miało krążyć po okręgu o średnicy 200 cm. Na rysunku 10.17 pokazano wycinek zarejestrowanej drogi, gdy urządzenia okrążają figurę jednokrotnie.



Rys. 10.17. Sterowanie niezależne czterech UAV – okrąg o średnicy 200 cm

## 11. Wnioski i uwagi

W rozprawie przedstawiono wyniki badań autorskich bezzałogowych jednostek latających (UAV) oraz autorskiego bezzałogowego systemu latającego (UAS). Przedstawiono budowę poszczególnych UAV, wynik badań komunikacji radiowej w ramach UAS, a także symulacje oraz testy algorytmów lotu w grupach do 5-ciu UAV.

Osiągnięto główny cel rozprawy tzn. wykonano kontrolowane loty z wykorzystaniem pięciu UAV. Opublikowane w rozprawie wyniki pozwalają na stwierdzenie, iż teza pracy *Możliwe jest zdecentralizowane sterowanie bezzałogowego systemu latającego z wykorzystaniem algorytmów deterministycznych* została udowodniona. Również tezy pomocnicze zostały potwierdzone.

W rozprawie przetestowano szereg autorskich algorytmów, pozwalających w sposób deterministyczny sterować UAS z uwzględnieniem warunku decentralizacji. Opracowane algorytmy pozwalają na otrzymanie identycznych wyników dla tych samych parametrów początkowych (położenie dronów), niezależnie od miejsca wykonywania obliczeń.

Opracowany algorytm doboru w pary urządzeń do punktów w formacji (np. okręgu, gwiazdy lub innego kształtu), przy założeniu nieprzecinania się trajektorii cząstkowych wewnątrz UAS oraz dążeniu sumy wszystkich dróg do minimum, działa zgodnie z założeniami. Symulacje wykazały, że spełnia on swoje zadanie także dla wielkości sieci rzędu 10 000 losowo wybranych (współrzędnych) dronów i punktów. W pracy wykazano, że przy zachowaniu odpowiedniej rozdzielczości urządzeń i punktów docelowych system działa stabilnie. Algorytm ten wykorzystywano przy starcie systemu, szukając optymalnego doboru urządzeń do punktów w formacji ( $D_{1:n} \rightarrow P_{1:n}$ ), a także za każdym razem gdy zmieniono formację lotu. Z wykorzystaniem tego algorytmu możliwe było wykonanie zmiennych w czasie formacji (np. z okręgu na kwadrat).

Algorytm dotyczący znalezienia drogi wyjścia z labiryntu z wykorzystaniem metod analitycznych działał poprawnie, znajdując quasi-optymalną drogę dla wszystkich badanych przykładów. Wykorzystywano go w przypadku, gdy pomiędzy urządzeniem (bądź grupą), a punktami docelowymi wykryto przeszkodę. Algorytm pomagał w wyznaczeniu alternatywnej trasy.

W badaniach napotkano na problemy związane z poprawnym określeniem zwrotu urządzenia. Sporą trudnością było znalezienie rozwiązania pozwalającego na poprawny pomiar zwrotu urządzenia względem dowolnego układu współrzędnych. W zbudowanym laboratorium rozbieżność pomiarowa dla jednego z czujników wynosiła nawet 30°. Alternatywne rozwiązania z wykorzystaniem dwóch mobilnych punktów nawigacji Marvelmind nie miały sensu, ze względu na znaczące zwiększenie masy urządzeń oraz ograniczenie maksymalnej liczby urządzeń UAS do dwóch. Wyznaczenie zwrotu z wykorzystaniem czujnika IMU również nie dawało zadowalającego rezultatu, ze względu na błąd spowodowany niedokładnością całkowania. Ostatecznie wykorzystując magnetometr HMC5883L podzielono laboratorium na 15 stref, wewnątrz których wykonano pomiar dla czterech kątów wzorcowych (45°, 135°, 225° i 315°). Pomiary wykonano na wysokości podłogi, jednego oraz dwóch metrów. Na podstawie uzyskanych wyników stwierdzono, że rozbieżność pomiarowa dla kąta 315° jest najmniejsza. Dla tej wartości wyznaczono funkcję aproksymującą drugiego rzędu mającą korygować wskazanie magnetometru. Rozwiązanie to działało poprawnie.

W ramach pracy zaprojektowano, zasymulowano oraz wykonano dwie konstrukcje bezzałogowych urządzeń latających. W obu konstrukcjach duży wkład miał druk 3D w technologii FDM. Prowadzone badania skupiały się głównie na możliwości wykorzystania materiałów ABS, PLA, PLAWOOD, PETG, TPU, NanoCarbon (PA 12 + CF) do prototypowania, późniejszej produkcji oraz w razie uszkodzeń, szybkiej naprawy ramy. Na podstawie przeprowadzonych analiz stwierdzono, że pod względem finansowym (w porównaniu do alternatywnych gotowych konstrukcji) wykorzystanie druku 3D ma sens. Kolejną zaletą jest szybkie dorobienie części, gdy ulegnie uszkodzeniu, w porównaniu do komercyjnych rozwiązań wymagających zamówienia elementu. Obecnie na rynek wchodzi wiele nowych materiałów, które swoją wytrzymałością dorównują alternatywnym rozwiązaniom frezowanym (CFRP, GFRP, aluminium).

Istotnym problemem było znalezienie systemu umożliwiającego pozycjonowanie urządzeń wewnątrz budynku. Standardowe systemy nawigacji satelitarnej, w warunkach zewnętrznych cechują się dokładnością na poziomie 2 – 3 metrów. Wewnątrz budynku badanie jest dodatkowo zakłócone, gdyż odbiornik GNSS nie potrafi nawiązać łączności z wymaganą liczbą satelitów, bądź sygnał z nich jest zniekształcony. Ostatecznie wykorzystano rozwiązanie firmy Marvelmind, które do określenia pozycji bazuje na ultradźwiękach. Podczas badań nie uzyskano deklarowanej przez producenta częstotliwości 16 Hz odświeżania pozycji. W trakcie badań osiągnięto dla jednego urządzenia około 10 Analiza zaprezentowanych trajektorii UAS wskazuje, iż dokładność systemu uzależniona jest od częstotliwości odświeżania pozycji. Wraz ze wzrostem liczby urządzeń wewnątrz UAS, system zmuszony był zmniejszać swoją prędkość w celu zapobiegania kolizji. Niemniej jednak, możliwe było, z wykorzystaniem autorskich algorytmów deterministycznych, zdecentralizowane sterowanie torem lotu pięciu jednostek latających wewnątrz bezzałogowego systemu latającego.

W trakcie pracy wystąpiło także szereg problemów technicznych, które należało rozwiązać. Badania związane z UAS są szczególnie problematyczne ze względu na duże straty w razie ewentualnej awarii. Najczęściej kończyły się one uszkodzeniami, wymuszając wielokrotne naprawy, serwisowanie i przeglądy urządzeń.

Realizacja pracy pozwoliła także na wytyczenie kierunków dalszych badań, które opisano poniżej:

- W ramach rozprawy wykonano znaczną liczbę badań mających na celu weryfikację pracy urządzeń. Zbudowany system daje duże możliwości badań pojedynczego UAV, oraz ich zbioru tzn. UAS. Planuje się zbadanie wpływu jednostki napędowej oraz wymiarów konstrukcji na zużycie energii, stabilność lotu oraz układ sterowania.
- Podczas sterowania wykorzystywano regulatory PID. Modularność systemu pozwala na wykorzystanie innych metod regulacji.
- Istotnym problemem był zbyt wolny system nawigacji lokalnej. Dalsze prace będą skupiać się na poszukiwaniu rozwiązań, które pozwolą na bardziej precyzyjne pomiary, np. nawigacja inercyjna, bądź system wizyjny.
- Podczas badań sieci radiowych stwierdzono, że dla sześciu członów sieci, system pracuje z częstotliwością 22 Hz zapewniając stabilną komunikację. Zwiększenie liczby członków UAS będzie wymagało dalszych badań systemu komunikacji.
- Wykonując UAS, oraz symulując ich ruch rozważano tylko przemieszczenie względem osi  $x$ ,  $y$  Parametr wysokości UAV był stały i deklarowany przez operatora sieci. Planowana jest aktualizacja systemu umożliwiająca urządzeniom ruch we wszystkich trzech osiach.

## 12. Wybrane pozycje ze spisu literatury rozprawy doktorskiej

- [1] Alarifi A., Al-Salman A., Alsaleh M., Alnafessah A., Al-Hadhrami S., Al-Ammar, M., Al-Khalifa H.: „*Ultra wideband indoor positioning technologies: Analysis and recent advances*”, Sensors, 2016, 16(707).
- [2] Bian Y., Zhang X., Wu Y., Zhao X., Liu H., Su Y.: „*Influence of prompt timing and messages of an audio navigation system on driver behavior on an urban expressway with five exits*”, Accident Analysis & Prevention, Volume 157, 2021.
- [3] Buckley M., Eagleson G., Silverman B.: „*The estimation of residual variance in nonparametric regression*”, Biometrika, 1988, pp. 189–199.
- [4] Dong Z.Y., Xu W.M., Zhuang H.: „*Research on ZigBee Indoor Technology Positioning Based on RSSI*”, Procedia Computer Science, Volume 154, 2019, pp. 424-429.
- [5] Gallo E., Barrientos L.: „*Reduction of GNSS-Denied inertial navigation errors for fixed wing autonomous unmanned air vehicles*”, Aerospace Science and Technology, Volume 120, 2022.
- [6] Gosiewski, Z., Kwaśniewski, K.: „*Time minimization of rescue action realized by an*

- autonomous vehicle*”, Volume 9, Issue 12, December 2020, Article number 2099, pp. 1-19.
- [7] Hattenberger G., Alami R., Lacroix S.: „*Autonomous configuration control for uav formation flight in hostile environments*”, IFAC Proceedings Volumes, Volume 40, Issue 15, 2007, pp. 451-456.
- [8] He J., Zhao H.: „*A new smart safety navigation system for cycling based on audio technology*”, Safety Science, Volume 124, 2020.
- [9] Junka S., Schröder W., Schrocka S.: „*Design of Additively Manufactured Wind Tunnel Models for Use with UAVs*”, Procedia CIRP, Volume 60, 2017, pp. 241-246.
- [10] Lupashin S., Hehn M., Mueller M.W., Schoellig A.P., Sherback M., D’Andrea R.: „*A platform for aerial robotics research and demonstration: The Flying Machine Arena*”, Mechatronics, Volume 24, Issue 1, February 2014, pp. 41-54.
- [11] Lygouras E., Gasteratos A.: „*A novel moving-base RTK-GPS-Based wearable apparatus for precise localization of humans in peril*”, Microprocessors and Microsystems, Volume 82, 2021.
- [12] Madgwick S.: „*An efficient orientation filter for inertial and inertial/magnetic sensor arrays*”, [http://x-io.co.uk/res/doc/madgwick\\_internal\\_report.pdf](http://x-io.co.uk/res/doc/madgwick_internal_report.pdf).
- [13] Madgwick S., Harrison A.J., Vaidyanathan R.: „*Estimation of IMU and MARG orientation using a gradient descent algorithm*”, 2011 IEEE International Conference on Rehabilitation Robotics, 2011, pp. 1-7.
- [14] Marut A., Wojtowicz K., Falkowski K.: „*ArUco markers pose estimation in UAV landing aid system*”, 2019 IEEE 5th International Workshop on Metrology for AeroSpace (MetroAeroSpace), 2019, pp. 261-266.
- [15] Mousavi S., Afghah F., Ashdown J.D., Turck K.: „*Use of a quantum genetic algorithm for coalition formation in large-scale UAV networks*”, Ad Hoc Networks, Volume 87, 2019, pp. 26-36.
- [16] Pacholski N.: „*Extending the sensor edge smart drone positioning system*”, Praca inżynierska, The University of Technology, Adelaide 2013.
- [17] Pecho P., Ažaltovič V., Kandra B., Bugaj M.: „*Introduction study of design and layout of UAVs 3D printed wings in relation to optimal lightweight and load distribution*”, Transportation Research Procedia, Volume 40, 2019, pp. 861-868.
- [18] Pehlivanoglu Y.V., Pehlivanoglu P.: „*An enhanced genetic algorithm for path planning of autonomous UAV in target coverage problems*”, Applied Soft Computing, Volume 112, 2021.
- [19] Samuel M., Nazeem N., Sreevals P., Ramachandran R., Careena P.: „*Smart indoor navigation and proximity advertising with android application using BLE technology*”, Materials Today: Proceedings, Volume 43, Part 6, 2021, pp. 3799-3803.
- [20] van der Heijden B., Ledergerber A., Gill R., D’Andrea R.: „*Iterative Bias Estimation for an Ultra-Wideband Localization System*”, IFAC-PapersOnLine, Volume 53, Issue 2, 2020, pp. 1391-1396.
- [21] Vashi Y., Anand R., Jayakrishna K., Rajyalakshmi G., Raj A.S.: „*Design and analysis of 3D printed UAV wheel*”, Materials Today: Proceedings, Volume 46, Part 17, 2021, pp. 8307-8312.
- [22] Wang Z., Cheng X.: „*Adaptive optimization online IMU self-calibration method for visual-inertial navigation systems*”, Measurement, Volume 180, 2021.
- [23] Xue X., Qin H., Lu H.: „*High-precision time synchronization of kinematic navigation system using GNSS RTK differential carrier phase time transfer*”, Measurement, Volume 176, 2021.
- [24] Yang T., Hu Y., Yuan X., Mathar R.: „*Genetic Algorithm based UAV Trajectory Design in Wireless Power Transfer Systems*”, 2019 IEEE Wireless Communications and Networking Conference (WCNC), 2019, pp. 1-6.
- [25] Zhou Z., Luo D., Shao J., Xu Y., You Y.: „*Immune genetic algorithm based multi-UAV cooperative target search with event-triggered mechanism*”, Physical Communication, Volume 41, 2020.

### Strony internetowe (stan z dnia 01.09.2022)

- [26] <http://www.dielectriccorp.com/downloads/thermoplastics/petg.pdf>
- [27] <https://www.u-blox.com>
- [28] [https://en.wikipedia.org/wiki/Real-time\\_kinematic\\_positioning](https://en.wikipedia.org/wiki/Real-time_kinematic_positioning)
- [29] <http://zrobsobiedrona.blogspot.com/2014/06/2-budujemy-quadcopter-jak-to-dziala.html>

### Publikacje z udziałem autora rozprawy doktorskiej

- [X1] Szywalski P.: „*Projekt i wykonanie bezzałogowego systemu latającego*”, Praca inżynierska, Politechnika Opolska, Opole, 01.2016.
- [X2] Szywalski P.: „*Opracowanie algorytmu autonomicznego lotu dla bezzałogowego systemu latającego*”, Praca magisterska, Politechnika Opolska, Opole, 07.2017.
- [X3] Szywalski P., Wajnert D.: „*Projekt i wykonanie bezzałogowego systemu latającego*”, Śląskie Wiadomości Elektryczne, R. XXIV, Nr 2/2017 (132), pp. 7-9.
- [X4] Szywalski P., Wajnert D.: „*Design of the autonomous flight algorithm for unmanned aerial system*”, Śląskie Wiadomości Elektryczne, R. XXV, Nr 3/2018 (139), pp. 4-7.
- [X5] Szywalski P.: „*Calculation of the quadrocopter frame deformation with using the finite element method*”, Zeszyty Naukowe Politechniki Opolskiej, z. 76, Nr 368/2018, pp. 105-107.
- [X6] Szywalski P.: „*Control algorithms for drone swarms*”, Zeszyty Naukowe Politechniki Opolskiej, z. 76, Nr 368/2018, pp. 107-111.
- [X7] Szywalski P., Wajnert D.: „*Possibility Analysis of the Location Measurement by Using the GPS Receiver and Barometric Altimeter*”, Pomiary Automatyka Robotyka, Nr 3/2018, pp. 33-39.
- [X8] Szywalski P., Waindok A.: „*Analysis of the quadrocopter class 130 frame deformation made with using 3D printing technology*”, Przegląd Mechaniczny, R. LXXVII, Nr 6/2018, pp. 39-44.
- [X9] Szywalski P., Waindok A.: „*Project of a navigation system designed for monitoring of an apartment with using a mobile robot*”, Śląskie Wiadomości Elektryczne, Nr 4-2019, pp. 8-11.
- [X10] Szywalski P., Waindok A.: „*Design and fabrication of a four-wheeled robot using reverse kinematics*”, Śląskie Wiadomości Elektryczne, Nr 4-2019, pp. 12-15.
- [X11] Szywalski P., „*Precise indoor navigation system for mobile Robots*”, Zeszyty Naukowe Politechniki Opolskiej, Nr 369/2019, pp. 109-111.
- [X12] Szywalski P., „*Review of navigation systems used in unmanned Aerial vehicles*”, Zeszyty Naukowe Politechniki Opolskiej, Nr 369/2019, pp. 111-113.
- [X13] Szywalski P., Waindok A., Tomczuk B.: „*Komunikacja radiowa grupy dronów w paśmie 433 MHz w obszarze linii wysokiego napięcia*”, Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej Nr 67/2019, pp. 69-73.
- [X14] Szywalski P., Waindok A., Tomczuk B.: „*Praca grupy dronów w pobliżu linii najwyższego napięcia*”, Przegląd Elektrotechniczny, R. 96 NR 3/2020, pp. 51-54.
- [X15] Szywalski P., Waindok A.: „*Practical aspects of design and testing unmanned aerial vehicles*”, Acta Mechanica et Automatica, Vol.14 No.1 (2020), pp. 50-58.
- [X16] Szywalski P., Waindok A.: „*Algorithm for Planning, Generating and Simulating Trajectories for a Group of Drones*”, IEEE Xplore: 2020 International Conference Mechatronic Systems and Materials (MSM) (2020).