# 4. Algorytm przeszukiwania z zabronieniami

## 4.1 Schemat algorytmu

Przydział dronów do poszczególnych punktów końcowych w celu jego optymalizacji w danym zakresie nie jest banalny. Dokonanie zmiany celu lotu pomiędzy zaledwie dwoma dronami może całkowicie zmienić wyniki symulacji, ze względu na możliwość zajścia kolizji. W dodatku przy wzroście liczby dronów zwiększa się również liczba możliwych kombinacji, a tym samym pracochłonność przeanalizowania każdej z nich. Dlatego konieczne jest zastosowanie algorytmu heurystycznego w celu zrównoważenia czasu oraz jakości otrzymanego rozwiązania. Na potrzeby niniejszego projektu wybrano algorytm przeszukiwania z zabronieniami – Tabu Search(*TS*).

Algorytm Tabu Search ma swoje korzenie na przełomie lat 60-tych i 70-tych. Znaną dzisiaj formę algorytmu zawdzięczamy profesorowi Fredowi Gloverowi, który sformułował ją w latach 80-tych [4.1]. Idea działania opiera się na przeszukiwaniu określonego sąsiedztwa w poszukiwaniu najlepszego rozwiązania. W rozpatrywanym przypadku będzie to wymiana punktów docelowych pomiędzy dwoma dronami. Następnie algorytm wykonuje kolejne „ruchy” czyli zmiany sąsiedztwa na najlepsze, które mogło zostać osiągnięte z poprzedniego sąsiedztwa. Najlepszy wynik osiągnięty podczas działania algorytmu oraz odpowiadający mu przydział są finalnym rozwiązaniem problemu.

**Algorytm Tabu Search** (na podstawie pseudokodu z artykułu [4.2])

**Krok 1**:

Wyznacz rozwiązanie początkowe *xpocz* i podstaw: *x* := *xpocz*, *xts* := *xpocz*, *Qts* := Q(*xpocz*), *CA*=*False*

**Krok 2**: dla *k*=1 do *K* wykonaj: (*K*- warunek stopu):

*x*(*i\*, j\**) = arg min;

*x*(*i’,j’*) = arg min ;

podstaw: *x* := *x*(*i\*, j\**);

Jeżeli < *Qts*  to *xts* := *x, Qts* := Q(*x*);

Jeżeli < *Qts*  to *xts* := *x*(*i’,j’*)*, Qts* := Q(*x*(*i’,j’*)), *CA*=*True* ;

Dla każdego (*i*,*j*) podstaw: ;

Jeżeli *CA*=*False*, to *KLT*(*i\**,*j\**) = *T*, inaczej *KLT*(*i’*,*j’*) = *T;*

Jeżeli *CA*=*False*, to *DLT(i\**,*j\**):= *DLT(i\**,*j\**)+1, inaczej *DLT(i’*,*j’*):= *DLT(i’*,*j’*)+1, *CA*=*False.*

Opis oznaczeń w pseudokodzie:

*CA* - zmienna określająca czy zastosowano kryterium aspiracji,

α – wartość współczynnika kary,

*KLT* - lista tabu zawierająca liczbę tur, w których określony krok jest zabroniony,

*DLT* – lista tabu zawierająca liczbę poszczególnych dokonanych kroków algorytmu,

*T* – liczba tur, w których dany krok znajduje się na liście *KLT*.

## 4.2 Pamięć krótko- i długoterminowa

W zależności od potrzeb możliwe jest zastosowanie algorytmu przeszukiwania z zabronieniami na kilka sposobów. W tym przypadku użyte zostaną algorytmy z pamięcią krótkoterminową oraz algorytm z pamięcią zarówno krótko- jak i długoterminową.

Pamięć krótkoterminowa (*KLT*) to lista tabu, która posiada po jednej pozycji dla każdego możliwego ruchu algorytmu. Domyślnie zapełniona jest zerami. Po każdym kroku algorytmu, najpierw od wszystkich niezerowych wartości odejmuje się jeden, a następnie na pozycji odpowiadającej wykonanemu ruchowi pojawia się wartość *T,* czyli określony w liczbie kroków okres znajdowania się ruchu na liście tabu. Algorytm sprawdza dla każdego ruchu czy nie znajduje się na *KLT* (czyli czy *KLT*(*i*,*j*)>0). Jeśli *KLT*(*i*,*j*)=0, to gdy ruch okaże się najlepszy, może być wykonany. Jeśli jednak *KLT*(*i*,*j*)>0, to ruch może zostać wykonany tylko jeżeli spełni warunek aspiracji.

Warunek aspiracji to swoisty wytrych, który umożliwia algorytmowi wykonanie niedozwolonego ruchu. Jeśli jego wynik okaże się lepszy nie tylko od najlepszego wyniku „dozwolonego” ruchu w tym kroku algorytmu, ale także od najlepszego wyniku w perspektywie wszystkich dotychczasowych kroków algorytmu, to kryterium aspiracji zostaje spełnione.

Główną różnicą pomiędzy zastosowanymi dwoma typami algorytmu jest zastosowanie bądź pominięcie pamięci długoterminowej (*DLT*). Podobnie jak lista *KLT* zawiera po jednej pozycji dla każdego ruchu. Jednakże, jej przeznaczeniem jest zliczanie liczby wykonania poszczególnych ruchów. Informacja ta jest wykorzystywana przy obliczaniu wyniku dla rozpatrywanego przydziału, bowiem do podstawowego wyniku dodaje się iloczyn wartości na odpowiedniej pozycji *DLT* i stałej wartości współczynnika kary, podzielonego przez ilość wykonanych iteracji algorytmu. Ma to na celu zapobieganiu wykonywaniu tych samych ruchów zbyt wiele razy. [4.2]

## 2.3 Stosowane algorytmy

Problem przydziału jest powszechnie spotykanym problemem optymalizacyjnym. Pojawia się on przy m.in. masowej produkcji, logistyce, a nawet zwykłym planowaniu dnia. W wielu przypadkach człowiek potrafi samodzielnie rozwiązać taki problem, szczególnie gdy nie jest zbyt skomplikowany. Są też sytuacje, w których złożoność jest na tyle trudna, że poza najtęższymi umysłami na planecie, nikt nie jest w stanie szybko podać rozwiązania. Wtedy z pomocą przychodzą właśnie algorytmy.

Wiele z tych problemów można przedstawić za pomocą tabel, grafów czy też funkcji liniowych. Popularnymi algorytmami i sposobami do ich rozwiązywania są w zależności od problemu: algorytm Dijkstry, algorytm węgierski, metoda Symplex, programowanie liniowe, algorytm aproksymacyjny itp.. [2.1]

Istnieją też tak bardzo skomplikowane problemy, że konieczne jest użycie algorytmów heurystycznych. Stosowane są szczególnie w rozwiązywaniu problemów, których rozwiązanie wymagałoby ogromnej złożoności obliczeniowej, bądź dużej ilości czasu. Ich właściwością jest bowiem skrócenie czasu potrzebnego na obliczenia, możliwym kosztem jakości rozwiązania. Wśród algorytmów heurystycznych możemy znaleźć takie algorytmy jak: algorytm przeszukiwania z zabronieniami (Tabu Search), algorytmy genetyczne oraz algorytm symulowanego wyżarzania. [2.2]

Przykładami użycia jednego z algorytmów heurystycznych – algorytmu przeszukiwania z zabronieniami – są następujące:

* planowanie tras [2.3],
* niepermutacyjny problem przypływowy z kryterium sumacyjnym [2.4],
* uogólniony problem przydziału [4.2].

## 1.2 Cel i zakres pracy

Niniejsza praca ma na celu określenie trajektorii roju dronów pomiędzy dwoma zadanymi konfiguracjami, przy zadanych ograniczeniach przestrzennych i technicznych oraz optymalizację przydziału punktów docelowych do dronów pod kątem ustalonej funkcji kryterialnej. Na potrzeby pracy zdefiniowano model matematyczny problemu w przestrzeni trójwymiarowej. Opracowano również algorytm przybliżony, z uwzględnieniem specyfiki problemu.

Zakres pracy obejmuje:

* Przegląd informacji związanych z dronami i algorytmami,
* Formalizację modelu matematycznego na podstawie uzyskanych informacji,
* Opracowanie algorytmu przybliżonego określającego optymalny przydział punktów docelowych do dronów pod kątem funkcji celu,
* Napisanie programu realizującego w ramach symulacji model matematyczny oraz programu realizującego algorytm,
* Przeprowadzenie eksperymentów obliczeniowych dla zbioru reprezentatywnych zadań testowych,
* Sformułowanie wniosków na temat efektywności algorytmu oraz własności problemu.

## 2.2 Problemy optymalizacyjne w sterowaniu dronami

Sterowanie dronami nie jest prostym zadaniem. Jak powszechnie wiadomo ich lot jest możliwy dzięki zamontowanymi na nich wirnikom. Ich ilość zależy głównie od producenta, jednakże najczęściej drony posiadają cztery wirniki. Tutaj niestety pojawia się pewien problem, O ile lot tylko w osi *z* (wysokości) nie wymaga szczególnych działań, to już lot w osiach *x* oraz *y* wymaga, aby wirniki miały różne prędkości obrotowe. To zaś może powodować różnicę pomiędzy zadaną trajektorią drona, a rzeczywistą trajektorią [2.5]. Dlatego konieczne jest aby zachowywać określony minimalny dystans pomiędzy dronami.

Dodatkowym czynnikiem przemawiającym za zachowywaniem dystansu jest skutek pracy wirników. Wytwarzanie siły jest możliwe tylko dzięki szybkiemu „wypychaniu” powietrza, dającemu potrzebny efekt odrzutu. Może to powodować nietypowe (turbulentne) ruchy powietrza zarówno powyżej jak i poniżej drona. Gdyby inny dron znalazłby się w takiej strefie, jego dalsza trajektoria mogłaby zostać w pewnym stopniu zaburzona, a w najgorszym przypadku kontrola nad nim byłaby niemożliwa i stałby się zagrożeniem dla innych dronów, a nawet mógłby się rozbić.

Kolejną przeszkodą jest ograniczona pojemność akumulatora. Im dłużej wirniki obracają się tym bardziej zapasy energii kurczą się, tym samym ograniczając maksymalny czas jaki może spędzić w powietrzu. W dodatku po przekroczeniu pewnej granicy poziomu energii maksymalna osiągalna prędkość obrotowa wirników może być nieosiągalna. Wymusza to wymianę bądź ładowanie akumulatora w odpowiednim czasie, a także ograniczenie czasu użycia drona.

Jeszcze innym problemem jest udźwig. Jest to szczególnie palący problem w użyciu dronów do celów transportowych. Technologia ta jest wprowadzana m. in. przez firmę Amazon [2.6]. Wymaga jednak użycia dronów, które są w stanie wytworzyć dużą siłę potrzebną do lotu. Poza tym należy także wziąć pod uwagę dodatkowy opór powietrza spowodowany zwiększeniem się powierzchni, co ponownie pochłania część wytworzonej siły.

## 3.1 Podstawowe parametry symulacji

Symulowanie lotu dronów wymaga określenia pewnych parametrów. Podstawą jest wyznaczenie przestrzeni symulacji. Tak jak w rzeczywistości jest ona trójwymiarowa. Jej długość, szerokość oraz wysokość jest określana przez operatora symulacji. Zakłada się, że przestrzeń ta jest wypełniona równomiernie powietrzem atmosferycznym o uśrednionych proporcjach poszczególnych jego składników. Ponadto ciśnienie atmosferyczne oraz gęstość powietrza uznaje się za jednakowe niezależnie od wysokości. Dodatkowo uznaje się, że panująca temperatura nie wpływa w żaden sposób na powietrze oraz pracę dronów.

Dodatkowym czynnikiem wprowadzonym do symulacji jest poruszanie się powietrza czyli wiatr. Określana jest jego prędkość względem osi *x* oraz *y* (względem osi *z* zawsze wynosi 0), która w trakcie trwania symulacji jest stała. Co więcej zakłada się, że ruch cząsteczek jest zawsze prostoliniowy, a nigdy nie jest turbulentny. Wyjątkiem jest przelot drona. Na potrzeby wzoru na opór aerodynamiczny zakłada się, że zaraz za dronem powietrze znajduje się w ruchu turbulentnym. To zaburzenie jednak trwa na tyle krótko, iż nie wpływa na inne drony (kluczowa jest tutaj strefa bezpieczeństwa drona o której więcej informacji poniżej).

Pozycję drona w przestrzeni symulacji określają współrzędne (*x*,*y*,*z*). Jest to dokładna pozycja jego środka ciężkości. Symulacja polega na określaniu kolejnych pozycji co podany okres czasu (okres próbkowania). Czas ten także jest wyznaczany przez operatora symulacji w zależności od oczekiwanej dokładności. Należy tutaj wspomnieć, iż w pewnym zakresie obliczeń rozwiązuje się równanie różniczkowe metodą Eulera. Zbieżność uzyskanego wyniku z wynikiem rzeczywistym jest tym lepsza im mniejszy jest okres próbkowania.

Z perspektywy operatora wyznacza się również siłę oraz maksymalną prędkość drona. Siła to nic innego jak maksymalna siła jaką są w stanie wytworzyć wirniki „wypychając” powietrze. Maksymalna prędkość to swoiste ograniczenie prędkości jaką mogą rozwinąć drony, które służy do testów właściwości symulacji.

## 3.2 Wyznaczanie kolejnych pozycji dronów

Podstawą sterowania dronami jest wyznaczenie przedziałów czasu i odległości, w których przyspieszają, hamują bądź lecą z jednostajną prędkością. Ruch drona można opisać jednym z trzech wariantów ruchu.

Pierwszym wariantem jest lot na dystansie pozwalającym rozwinąć prędkość maksymalną. Dron przyspiesza do tejże prędkości, a następnie leci zachowując jej wartość. Gdy znajdzie się w odległości od punktu docelowego równej wyznaczonej drodze hamowania, dron rozpoczyna hamowanie i zatrzymuje się idealnie w wyznaczonym miejscu.

Drugim wariantem jest lot na dystansie zbyt krótkim aby rozwinąć maksymalną prędkość, ale na tyle długim aby przemieszczać się dłużej niż jeden okres próbkowania. Dron przelatuje wtedy całą odległość z przyspieszeniem bądź z opóźnieniem. Dla potrzeb sterowania obliczany jest przybliżony dystans przyspieszania i hamowania. Następnie jest wykonywany zgodnie z uzyskanymi wartościami. Ze względu na możliwe różnice pomiędzy faktycznym miejscem zatrzymania, a punktem docelowym (wynikające z przybliżenia) wprowadza się pewien margines błędu. Jego istnienie jest dopuszczalne, bowiem dron zawsze znajdzie się dostatecznie blisko, aby punkt docelowy został objęty co najmniej strefą bezpieczeństwa drona. Uznaje się więc, że w rzeczywistości dron delikatnie skorygowałby swoją pozycję do zadanej. Nie wpływa to znacznie na kryteria funkcji celu, dlatego pomija się ten fakt w symulacji i lot drona zostaje sklasyfikowany jako ukończony w punkcie docelowym.

Ostatnim wariantem jest najprostszy wariant: dron osiąga punkt docelowy w jednym okresie próbkowania. Nie wymaga to żadnych dodatkowych obliczeń.

Na wstępie należy pochylić się nad siłami wpływającymi na lot drona. Wynikiem dodania tych sił wektorowo jest wektor wypadkowy siły ., której kierunek powinien być identyczny do kierunku zadanej trajektorii. Do celów obliczeniowych wygodniejsze jest operowanie na jej składowych: , i . Pierwszą ważną siłą wpływającą na wektor wypadkowy jest siła ciężkości drona, którą oblicza się za wzoru 3.1.

– siła ciężkości drona

– masa drona

– przyspieszenie ziemskie = 9.8

Jej zwrot jest zawsze przeciwny do zwrotu osi *z* i jest prostopadły do innych osi, więc będzie wpływała jedynie na składową . Inaczej jest z siłą oporu aerodynamicznego. Oblicza się ją ze wzoru 3.2.

– opór aerodynamiczny

-współczynnik oporu aerodynamicznego drona

– gęstość powietrza =1,2

– pole przekroju poprzecznego drona

-prędkość drona

Na siłę oporu składa się kilka zmiennych. Współczynnik oporu aerodynamicznego oraz pole przekroju drona zależą od kierunku przemieszczania się powietrza względem drona. Dla uproszczenia uznaje się wartości te za takie same (różnice nie są spektakularnie duże).

Największy problem stanowi jednak prędkość, gdyż nie jest to prędkość drona ale powietrza. Chociaż w bezwietrznej sytuacji wektor prędkości powietrza różniłby się od wektora prędkości drona jedynie przeciwnym zwrotem, to przy wietrze należy dodać jeszcze jeden wektor. Dopiero wartość tego wektora może zostać wykorzystana we wzorze 3.2. Podstawiając wszystkie zmienne do wzoru w wyniku otrzymujemy wartość wektora siły oporu, który ma kierunek i zwrot zgodny z wektorem prędkości powietrza. Do celów obliczeniowych wektor ten zostaje rozłożony na wektory *Dx, Dy* i *Dz*.

Teraz należy przyjrzeć się samej sile . Kierunek wektora siły wypadkowej musi być zgodny z wyznaczoną trajektorią prostoliniową. W związku z tym należy odpowiednio skierować siłę drona . Znając współrzędne punktu początkowego (x1,y1,z1) oraz punktu docelowego (x2,y2,z2) można opierając się na właściwościach rozkładu wektora na składowe wyznaczyć równanie:

Mając obliczone wszystkie znane siły można zapisać następujący układ równań:

Gdzie *,* ito składowe wektora . Po przekształceniu układu równań 3.4 otrzymuje się:

Korzystając z twierdzenia pitagorasa można zapisać siłę drona F jakoo:

Pamiętając, że wszystkie wektory zostały pogrupowane względem równoległości do danej osi oraz uwzględnia się ich zwrot względem niej można zapisać:

Za pomocą proporcjonalności składowych siły wypadkowej można podstawić jedną ze składowych sił w miejsce pozostałych jeśli nie jest równa 0. Przykładowo dla składowej siły wzór prezentowałby się następująco:

Otrzymane równanie ma jedynie jedną niewiadomą (). Po jego uporządkowaniu otrzymuje się równanie kwadratowe, które da się rozwiązać za pomocą tzw. delty. Otrzymaną wartość wstawia się do równania 3.3 i w rezultacie otrzymuje się brakujące składowe siły wypadkowej. Za pomocą twierdzenia Pitagorasa (podobnego do równania 3.6) otrzymuje się finalną wartość siły wypadkowej jaka powoduje przyspieszenie drona.

Do kolejnych obliczeń potrzebne jest przyspieszenie jakie nadaje siła . Z przekształcenia drugiej zasady dynamiki Newtona wynika:

– wektor przyspieszenia drona

– wypadkowa siła

– masa drona

Obliczone przyspieszenie w rzeczywistości zmieniałoby się w czasie, bowiem każda zmiana prędkości (czyli efekt przyspieszenia) powoduje zmianę oporu aerodynamicznego, co zmienia wartość siły wypadkowej powodującej przyspieszenie. Mamy więc do czynienia ze swoistym równaniem różniczkowym. Na potrzeby projektu oblicza się przyspieszenie co pewien okres próbkowania, tj. korzysta się z metody Eulera na obliczanie tego typu równań.

Do obliczania prędkości i drogi jaką pokonuje dron w czasie jednej próbki stosuje się następujące wzory:

– prędkość chwilowa drona

– prędkość drona w poprzedniej próbce

– przyspieszenie drona

– czas próbkowania [s]

droga pokonana przez drona w czasie jednej próbki [m]

Tutaj sprawa staje się banalnie prosta bowiem za pomocą twierdzenia Talesa oraz twierdzenia Pitagorasa można obliczyć dokładną pozycję drona w kolejnym kroku symulacji jak widać na rysunku 3.1.

*x1*

*x2*

*Sx*

*S*

*Rysunek 3.1 Rozkład odległości z trzech wymiarów na wymiar x*

## 3.3 Kolizje

Każdy dron posiada własną strefę bezpieczeństwa wyznaczoną względem wszystkich osi (np. . Pojawienie się innego drona w tej strefie uznaje się zawsze jako kolizję, niezależnie od tego czy fizyczna kolizja rzeczywiście mogła nastąpić. Należy bowiem brać pod uwagę także inne niebezpieczeństwa wynikające z bliskości dronów opisane w rozdziale 2.2.

Funkcja sprawdzająca czy dochodzi do kolizji otrzymuje pełną listę zawierającą kolejne pozycje dronów w każdej próbce symulacji. Następnie wszystkie pozycje w danym kroku sprawdza się pod kątem odległości we wszystkich trzech wymiarach. Celem uniknięcia powtarzania porównań między tymi samymi pozycjami, wpierw sprawdza się ewentualne kolizje pomiędzy pierwszym dronem i wszystkimi pozostałymi, później zaś pomiędzy drugim, a tymi których numer porządkowy jest wyższy itd.. Jeśli odległości we wszystkich wymiarach będą mniejsze od wyznaczonych dla strefy, to funkcja oznacza takie zbliżenie jako kolizję.

## 3.4 Funkcja celu

Bardzo ważnym pod względem optymalizacji symulacji jest określająca ją funkcja celu. Jej wartość uzyskuje się z następującego równania:

– czas potrzebny na zajęcie punktów docelowych przez wszystkie drony

*k* – liczba kolizji pomiędzy dronami

*d* – średnie oddalenie dronów podczas symulacji [m]

*a,b,c –* mnożniki poszczególnych zmiennych

Każde z wymienionych kryteriów ma swój własny mnożnik. Jest on wyznaczany odgórnie w zależności od potrzeb symulacji i testów. Najważniejszym z punktu widzenia realizmu jest kryterium kolizji, które powinno praktycznie uniemożliwiać zaliczenie symulacji do jednych z najlepszych. Dlatego mnożnik tego kryterium musi być w porównaniu z pozostałymi dużo większy.

Kryterium czasu trwania zmiany pozycji wszystkich dronów jest szczególnie ważne w kontekście strat energii oraz oszczędności czasu. Co prawda w symulacji zakłada się, że energia akumulatorów dronów pozostaje niezmienna, ale w przypadku użycia symulacji do rzeczywistego wykorzystania kryterium to staje się niezwykle ważne. Jego mnożnik należy więc dostosować w zależności od preferencji.

Można bowiem wyżej cenić kryterium oddalenia dronów od siebie. Jest to szczególnie pożądane jeśli chodzi o estetykę, gdyż donioślej wygląda lot dronów daleko od siebie niż stłoczenie na bardzo małej przestrzeni. Mnożnik tego kryterium musi więc być odpowiednio dostosowany względem mnożnika kryterium czasu trwania.

Ideą istnienia funkcji celu jest jej minimalizacja. Zadanie znalezienia takiego przydziału dronów do punktów docelowych, aby wynikająca z cech symulacji funkcja celu była jak najniższa, należy do zastosowanego algorytmu.

## 2.1 Sposoby użycia dronów

Pędzący rozwój technologiczny pozwala w dzisiejszych czasach na realizowanie projektów, które do tej pory były niemożliwe, bądź skrajnie trudne do wykonania. Pomimo bogatej historii bezzałogowych statków powietrznych, dopiero w ubiegłej dekadzie zyskały one uznanie mas i stały się niezwykle popularne. Ich powszechność z pewnością jest katalizatorem wielu inspiracji do wykorzystania ich w praktyce, a nie tylko do zabawy.

Najczęściej wykorzystuje się zdolność dronów do obserwowania obiektów z dużej wysokości. Jest to szczególnie ważne, gdy obserwowany obszar ma znaczną powierzchnię, bądź jest niezwykle trudno dostępny. Dlatego idealnie nadają się do pomocy w chociażby leśnictwie. Umożliwiają wtedy m. in. obserwację migracji zwierząt, badanie wilgotności, tropienie kłusowników oraz lokalizację martwych drzew. W przypadku straży pożarnej stają się doskonałym środkiem do kontrolowania pożarów (np. lasów). [2.8]

Docieranie do trudno dostępnych miejsc można wykorzystać także w inny sposób. Rozwój technologii pozwala obecnie na przesyłanie dronami niewielkich przesyłek, tam gdzie dotarcie drogą lądową byłoby niemożliwe lub niezwykle czasochłonne. Ponownie należy przytoczyć przykład leśnictwa, gdyż szczepienie zwierząt zrzucanymi na ziemię szczepionkami ukrytymi w pożywieniu jest doskonałym na to przykładem. Jednym z bardziej niecodziennych rozwiązań jest użycie drona do rzucenia koła ratunkowego tonącemu. Skuteczność takiego ratunku byłaby co najmniej wątpliwa. [2.8]

Niektóre wielkie koncerny nie zamierzają poprzestać na użyciu bezzałogowych statków powietrznych jedynie w ekstremalnych sytuacjach. Firma Amazon w 2013 roku ogłosiła plany co do wprowadzenia poczty lotniczej. Polegałaby na dostarczeniu przez drona produktu z magazynu pod drzwi zamawiającego. Podobnie swoje usługi w tej materii zaczyna wprowadzać firma Google. Wygląda więc na to, że nieuniknionym elementem przyszłości będzie przesyłanie paczek drogą powietrzną. Można jedynie zastanowić się jak bardzo zatłoczona będzie wtedy przestrzeń powietrzna. [2.7]

Jednakże nie tylko cele praktyczne muszą determinować wykorzystanie dronów. Jednymi z najchętniej oglądanymi na całym świecie spektaklami są ceremonie otwarcia igrzysk olimpijskich. Nie dziwi więc, że w każdej kolejnej edycji organizatorzy stają na głowie aby jak najbardziej je uatrakcyjnić, pomagając sobie najnowszymi nowinkami technologicznymi. Nie inaczej było na ceremoniach otwarcia igrzysk w Pjongczangu, czy też w Tokio. W samym Pjongczangu do niezwykłego spektaklu świetlnego użyto 1 218 dronów, które formując przeróżne konstelacje, za pomocą zainstalowanych źródeł światła, formowały kształty, które z pewności zachwycały zgormadzonych widzów.



*Rysunek 2.1 Ceremonia otwarcia zimowych igrzysk olimpijskich w Pjongczang* [2.9]

## Wprowadzenie

Bezzałogowe statki powietrzne (szerzej znane jako drony) to, jak sama nazwa wskazuje, statek powietrzny niewymagający obecności załogi na jego pokładzie oraz niemogący zabrać na swój pokład pasażerów [1]. W zależności od przeznaczenia mogą być duże jak samoloty pasażerskie lub tak małe, że można schować je do kieszeni. Dzięki ich znakomitej mobilności mogą być wykorzystywane do wielu zadań (więcej w rozdziale 2.1).

Pomimo, iż drony dopiero od niedawna są powszechnie spotykane, to pierwsze konstrukcje pojawiły się jeszcze w pierwszej połowie XX wieku. W 1920 roku Etienne Oehmichen stworzył kwadrokopter, który przeleciał rekordowe na tamte czasy 360 metrów [2.7]. Nie był to jednak dron, bowiem Oehmichen osobiście nim sterował. Konstrukcja ta przypominała rozwiązaniami technicznymi obecne modele dronów. Wraz z rozwojem technologii ich produkcja stawała się coraz łatwiejsza. W ostatniej dekadzie stały się szczególnie popularne, czemu sprzyjała duża różnorodność modeli dostępnych na rynku.

Obraz zawierający tekst, samolot, transport

Opis wygenerowany automatycznie

*Rysunek 1.1 Kwadrokopter Etienne’a Oehmichena z 1922r.*

Rosnąca z roku na rok liczba wyprodukowanych dronów pozwala na wykorzystanie ich na dużą skalę. Przykładem mogą być ceremonie otwarcia Igrzysk Olimpijskich w Pjongczangu w 2018 roku oraz w Tokio w 2021r. W samym Pjongczangu użyto 1 218 dronów do stworzenia spektaklu świetlnego, którego tłem było nocne niebo.



*Rysunek 1.2 Ceremonia otwarcia Letnich Igrzysk Olimpijskich w Tokio w 2021 r.* [2]

Wykorzystanie większej liczby dronów do celów artystycznych wymaga rozwiązania szeregu problemów. Należy odpowiednio wytyczyć trajektorie wszystkich dronów, aby nie dopuścić do kolizji oraz aby każdy dron znalazł się w swoim punkcie docelowym. Ponadto na drony wpływają warunki atmosferyczne, a czas ich lotu bez ładowania akumulatora jest ograniczony. W związku z tym aby prawidłowo zaprojektować taki spektakl trzeba wziąć pod uwagę wiele kryteriów i zmiennych. Duża czasochłonność obliczeń jakie należałoby wykonać wymusza użycia do tego celu jednostek obliczeniowych. W ramach niniejszego projektu zaproponowany zostanie sposób na rozwiązanie tego problemu.

# Opis aplikacji

## Środowisko programistyczne

Do implementacji opisanych w poprzednich rozdziałach modelu matematycznego oraz algorytmu przeszukiwania z zabronieniami wykorzystano program Spyder. Oprogramowanie to zostało wydane na otwartej licencji MIT, dzięki czemu jest ono darmowe i powszechnie dostępne [15]. Program ten został wybrany dzięki dużej łatwości sprawdzania wartości poszczególnych zmiennych bez potrzeby zamieszczania polecenia ich wypisania w kodzie. Pozwala to na szybsze znajdywanie błędów w implementacji oraz na wygodne analizowanie uzyskanych wyników.

Obraz zawierający tekst

Opis wygenerowany automatycznie

*Rysunek 5.1 Zrzut ekranu programu Spyder*

Implementacja została wykonana za pomocą języka Python, który jest obecnie bardzo często używany. Jest to niezwykle przyjazny dla użytkownika język, który automatycznie przypisuje typy zmiennym oraz pozwala na zapisanie w jednej linijce instrukcji, które w innych językach potrzebują dużo więcej miejsca.

## Proces powstawania i testowania kodu

Całość kodu została zawarta w trzech plikach Pythona: *Zmienne*, *Symulacja* oraz *Algorytm*. W pliku *Zmienne* zostały umieszczone wszystkie zadawane zmienne wpływające na wygląd symulacji. Są to m. in. wymiary przestrzeni symulacji, liczba dronów, wymiary strefy bezpieczeństwa drona, prędkość wiatru w poszczególnych osiach oraz maksymalna siła wytwarzana przez drona. Tutaj została również zdefiniowana klasa Drone, która przechowuje wszystkie informacje dotyczące pojedynczego drona (np. prędkość, punkt docelowy, akyualna pozycja itp.). Ponadto w liku tym znajdują się również funkcje wykorzystywane w klasie Drone (np. sprawdzanie kierunku lotu), oraz funkcje przygotowujące losowe punkty startowe i punkty docelowe.

Plik o nazwie *Symulacja* przechowuje funkcje obliczeniowe. Ich zadaniem jest wykrycie i realizacja odpowiedniego wariantu sterowania, obliczanie kolejnych pozycji dronów zgodnie z zadanym modelem matematycznym, oraz obliczanie kryteriów potrzebnych do funkcji celu.

Ostatni plik (*Algorytm*) jest najważniejszy, ze względu na swoją zawartość. To w nim znajduje się oraz jest wywoływana funkcja realizująca algorytm. Do jej działania wykorzystywana jest zawartość pozostałych plików.

Podczas programowania dokonywano po skończeniu kolejnego etapu prac testów zachowania programu. Wpierw sprawdzone zostało działanie funkcji losowo dobierającej punkty startowe oraz docelowe. Dopiero po jej pozytywnej weryfikacji możliwe było kodowanie modelu matematycznego.

Kodowanie modelu matematycznego było podzielone na dwa etapy ze względu na jego dużą złożoność. W pierwszym etapie stworzono funkcje obliczające kolejne pozycje, przy czym przyspieszenie zostało podane jako stała wartość. Dzięki takiemu rozwiązaniu weryfikacja prawidłowości lotu dronów była dużo prostsza. Tutaj pojawił się problem dotyczący niedokładności sterowania dronem (dokładnie chodziło o zbyt szybkie hamowanie). Został on rozwiązany poprzez wprowadzenie maksymalnego błędu odległości, zależnego od maksymalnej siły drona. Jeśli dron znajduje się wystarczająco blisko, to za jego kolejną pozycję uznaje się pozycję docelową. Jest to dopuszczalne, gdyż dron wykorzystuje do hamowania maksymalną siłę ciągu, więc w rzeczywistości jej częściowe zmniejszenie pozwoliłoby dronowi na zatrzymanie się dokładnie w wyznaczonym miejscu.

W drugim etapie zbudowane zostały funkcje obliczające przyspieszenie zgodnie ze wzorami z rozdziału 3.2. Ze względu na różniczkowy charakter równań przyspieszenie jest obliczane jedynie co krok symulacji (czyli co określony interwał czasowy w symulacji). Podczas testów zauważony został problem zbyt małej siły ciągu drona. W skrajnie trudnych przypadkach siła drona nie wystarczała do pokonania siły ciężkości i oporów ruchu dla danej prędkości maksymalnej. To zaś powodowało pojawienie się zespolonych rozwiązań równania kwadratowego. W celu wykrywania tego problemu stworzony został specjalny wyjątek zatrzymujący całą symulację i wskazujący na ten problem.

Ostatnim brakującym ogniwem była implementacja algorytmu. Została ona wykonana najpierw dla algorytmu TS z pamięcią krótkotrwałą. Później zaś na jej podstawie, po dodaniu kilku niezbędnych elementów, powstał algorytm TS z pamięcią krótko- i długotrwałą.

Podczas pisania całego kodu pojawiało się kilka mniejszych problemów, które zostały rozwiązane. Często wynikały one z błędów w składni kodu bądź z błędów literowych. Pojawiały się również błędy niszczące wyniki symulacji, poprzez błędne obliczenia w jednej z funkcji. Mimo to udało się uzyskać prawidłowo działający program realizujący model matematyczny oraz algorytm.

## Optymalna liczba dronów

Projektowanie pokazu z udziałem dronów wymaga określenia pewnych założeń wstępnych. Wśród nich są m.in. liczba dronów tworzących konstelacje oraz rozmiary przestrzeni, w której poruszają się drony. Założenia te są ze sobą powiązane, bowiem każdy dron potrzebuje objętość przestrzeni równą co najmniej swojej objętości. Zakładając, że dron ma objętość 0,1 m3 hipotetycznie można założyć, że w 1m3 przestrzeni zmieści się 10 dronów. Jednakże w takim przypadku jakikolwiek ruch byłby niemożliwy. Stąd rodzi się pytanie: Jaka objętość powinna przypadać na jednego drona, aby możliwy był ich bezkolizyjny lot?

Najlepszym sposobem na znalezienie odpowiedzi na to pytanie jest wykonanie szeregu testów za pomocą symulacji oraz analiza ich wyników. Jest to z pewnością lepsze wyjście niż wykonywanie testów na prawdziwych dronach w realnych warunkach.

Pierwsze testy wykonano z udziałem od dwóch do dziesięciu dronów w przestrzeni o wymiarach 10m x 10m x 10m czyli o objętości 1000m3. Strefa bezpieczeństwa każdego pojedynczego drona wynosiła 1m x 1m x 2m. W ostatniej osi (osi *z*) strefa została powiększona, aby drony nie wpadały w turbulencje (rozdział 3.1). W funkcji celu używane było kryterium czasu trwania przelotu (z mnożnikiem równym 1) oraz liczba kolizji (z mnożnikiem równym 1000). Dzięki temu każde uzyskane rozwiązanie, w którym dochodzi do kolizji ma wartość większą niż 1000, zaś to bez kolizji ma wartość mniejszą niż 1000. Interwał czasowy obliczania pozycji dronów wynosi 0,1s. Pozycje początkowe i końcowe były przydzielane losowo. Warunkiem zakończenia działania algorytmu jest nieznalezienie lepszego rozwiązania przez liczbę kroków równą liczbie dronów w symulacji pomniejszonej o 1. Istotne parametry symulacji przedstawia tabela 6.1.

*Tabela 6.1 Istotne parametry testowanych symulacji*

|  |  |  |  |
| --- | --- | --- | --- |
| Wymiary przestrzeni [m] | 10 x 10 x 10 | Siłą nośna drona [N] | 10 |
| Liczba dronów | W zależności od badania | Masa drona [kg] | 1 |
| Prędkość maksymalna | 1 | Użyty typ algorytmu | KLT, KLT+DLT |
| Interwał czasowy symulacji [s] | 0,1 | Wartości mnożników kryteriów funkcji celu | a=1  b=1000  c=0 |
| Prędkość wiatru (względem osi x i y) | x: 0  y: 0 |

Niestety po wykonaniu testów okazało się, że część wyników jest zaburzona, bowiem funkcje zapobiegjące przydzielaniu tych samych pozycji początkowych i docelowych nie spełniały swoich funkcji, w wyniku czego zdarzały się symulacje, w których dwa, a nawet więcej dronów miało ten sam punkt docelowy, co powodowało w każdym przypadku wykrycie kolizji. Dlatego w analizie wyników nie uwzględniano tych przypadków.

*Wykres 6.1 Procent najlepszych symulacji (w każdym kroku algorytmu z KLT) z kolizjami.*

Wykres 6.1 powstał na podstawie testów wykonywanych za pomocą algorytmu TS z pamięcią krótkotrwałą. Zliczane były wszystkie najlepsze uzyskane wyniki w każdym kroku algorytmu. Mimo to za każdym razem algorytm znajdował rozwiązanie, w którym kolizje nie występowały. Należy jednakże zauważyć, że wraz ze wzrostem liczby dronów zwiększa się liczba rozwiązań, które zawierają w sobie kolizje. Należy się więc spodziewać, że przy pewnej liczbie dronów uzyskanie bezkolizyjnego rozwiązania będzie niemożliwe. Podobne wyniki zostały uzyskane przy użyciu pamięci krótko- i długotrwałej algorytmu.

*Wykres 6.2 Procent najlepszych symulacji (w każdym kroku algorytmu z KLT+DLT) z kolizjami.*

Jak można zauważyć na wykresie 6.2, algorytm używający oba typy pamięci poradził sobie znakomicie. Uzyskana zależność jest niemalże liniowa, a w dodatku uzyskane wyniki są relatywnie lepsze niż przy użyciu tylko pamięci krótkotrwałej. Podobnie jak w poprzednim przypadku wszystkie ostateczne rozwiązania znalezione przez algorytm były bezkolizyjne.

Przy poprzednich założeniach nie sposób było określić maksymalną liczbę dronów, jaka może poruszać się w ograniczonej przestrzeni. Powodem była zbyt duża przestrzeń oraz zbyt mała liczba dronów. Niestety zwiększenie liczby dronów wiązałoby się ze znacznym wydłużeniem czasu obliczania rozwiązań. Dlatego wykonano drugi zestaw testów skupiający się jedynie na ostatecznych rozwiązaniach. Tym razem przestrzeń została ograniczona do wymiarów 5m x 5m x 5m. Parametry tych symulacji prezentuje tabela 6.2.

*Tabela 6.2 Istotne parametry testowanych symulacji*

|  |  |  |  |
| --- | --- | --- | --- |
| Wymiary przestrzeni [m] | 5 x 5 x 5 | Siłą nośna drona [N] | 20 |
| Liczba dronów | W zależności od badania | Masa drona [kg] | 1 |
| Prędkość maksymalna | 1 | Użyty typ algorytmu | KLT+DLT |
| Interwał czasowy symulacji [s] | 0,1 | Wartości mnożników kryteriów funkcji celu | a=1  b=1000  c=0 |
| Prędkość wiatru (względem osi x i y) | x: 0  y: 0 |

*Wykres 6.3 Liczba ostatecznych rozwiązań z kolizjami na 10 rozwiązań.*

Z wykresu 6.3 wynika, że rozwiązania z kolizjami pojawiły się dopiero przy sześciu dronach. Dla siedmiu dronów udało się znaleźć jedynie bezkolizyjne rozwiązania, co może oznaczać, że możliwe jest wykorzystanie większej ilości dronów przy odpowiednich warunkach (głównie chodzi o zestaw pozycji).

Podsumowując testy wykazały, że przy określonych założeniach bezpiecznym stosunkiem liczby dronów do objętości przestrzeni jest jeden dron na 25 m3. Jeśli zaś chodzi o stosunek strefy bezpieczeństwa drona do objętości przestrzeni jest to 1m3 na 12,5m3.Wynika z tego, że dla pierwszych założeń (przestrzeni 10m x 10m x 10m) dopuszczalną liczbą dronów byłoby 40 dronów. Należy też zaznaczyć, że możliwe jest użycie większej liczby dronów, ale tylko w wybranych przypadkach.

Do dalszych testów używano tylko algorytmu z oboma typami pamięci, ze względu na jego skuteczniejsze działanie.

## Prędkość maksymalna dronów

Kolejną ważną zmienną ustalaną na potrzeby pokazu jest zadana maksymalna prędkość dronów. Zgodnie z założeniami drony rozpędzają się używając maksymalną siłę ciągu do prędkości maksymalnej, a następnie kontynuują lot z tą prędkością aż do pozycji, od której muszą rozpocząć hamowanie. Większa prędkość przelotowa oznacza równocześnie krótszy czas potrzebny do osiągnięcia celu. Aby zbadać zależność czasu od maksymalnej prędkości, wykonany został szereg symulacji o tych samych parametrach (łącznie z punktami początkowymi i docelowymi), w których zadawano różne prędkości maksymalne. Dokładne wartości parametrów zawiera tabela 6.3.

*Tabela 6.3 Istotne parametry testowanych symulacji*

|  |  |  |  |
| --- | --- | --- | --- |
| Wymiary przestrzeni [m] | 100 x 100 x 100 | Siłą nośna drona [N] | 10 |
| Liczba dronów | 10 | Masa drona [kg] | 1 |
| Prędkość maksymalna | W zależności od badania | Użyty typ algorytmu | KLT+DLT |
| Interwał czasowy symulacji [s] | 0,1 | Wartości mnożników kryteriów funkcji celu | a=1  b=1000  c=20 |
| Prędkość wiatru (względem osi x i y) | x: 0,1  y: -0,1 |

*Wykres 6.4 Czas potrzebny dronom na osiągnięcie punktów docelowych w funkcji maksymalnej prędkości*

Analizując wykres 6.4 można wywnioskować, że niekoniecznie zadanie jak najwyższej prędkości maksymalnej będzie najkorzystniejsze. Po pierwsze trzeba zauważyć, iż czasu przelotu od około 13 nie zmienia się aż tak bardzo, jak przy niższych prędkościach. Przy prędkościach powyżej 28 różnice są minimalne. Wpływa na to fakt, iż drony muszą rozpędzać się do zadanej prędkości oraz hamować, co przy niezmiennej maksymalnej sile ciągu wymaga pewnego czasu.

Należy pamiętać, że wyższe prędkości oznaczają większe opory powietrza, a co za tym idzie większe pobory mocy potrzebnej do wytwarzania siły ciągu. Dlatego przy wyborze prędkości maksymalnej, poza ograniczeniami technicznymi, należy również wziąć pod uwagę zwiększony pobór mocy. Korzystając z wykresu 6.4 można więc zawęzić zakres rozpatrywanych prędkości do maksymalnie 13, gdyż wyższe prędkości wpływają minimalnie na czas przelotu wszystkich dronów, ale generują większe koszty energetyczne.

Ciekawym aspektem są rozwiązania uzyskane podczas opisanych testów. O ile część dronów zawsze otrzymywała te same pozycje docelowe to kilka z nich otrzymywało jedną z pozostałych pozycji. Pozycje początkowe były punktami na odwodzie prostokąta o środku w środku przestrzeni, zaś pozycje docelowe leżały na przekątnej zadanej przestrzeni. Prawdopodobnie to właśnie symetryczność obu konstelacji powodowała te subtelne różnice (o ile nie była to kwestia niedoskonałości algorytmu).

## Wyrównywanie czasu lotu

Każdy pokaz z udziałem dronów niesie ze sobą również walory artystyczne. Z pewnością lepiej wygląda pokaz, w którym zmiana pozycji wszystkich dronów jest wykonywana w niemal identycznym czasie, niż gdy część dronów osiąga swoje pozycje niemal natychmiast, a pozostałe przelatują niemal na drugi koniec przestrzeni. W tym celu wykonana została drobna korekta w funkcji celu. Zamiast kryterium całkowitego czasu trwania przelotu wszystkich dronów wprowadzona została różnica pomiędzy całkowitym a najkrótszym czasem. Dzięki temu możliwe jest poszukiwanie rozwiązania, w którym drony zmieniają pozycje w sposób zbliżony do synchronicznego.

W celu zbadania działania algorytmu pod kątem zmniejszania różnicy w czasie lotu wykonanych zostało 10 testów z tymi samymi parametrami (tabela 6.4), lecz innymi zestawami pozycji. Do zestawienia wzięte zostały wszystkie najlepsze wyniki uzyskane w krokach algorytmu, aby widoczna była różnica pomiędzy najgorszym, średnim i najlepszym rozwiązaniem.

*Tabela 6.4 Istotne parametry testowanych symulacji*

|  |  |  |  |
| --- | --- | --- | --- |
| Wymiary przestrzeni [m] | 100 x 100 x 100 | Siłą nośna drona [N] | 20 |
| Liczba dronów | 10 | Masa drona [kg] | 1 |
| Prędkość maksymalna | 5 | Użyty typ algorytmu | KLT+DLT |
| Interwał czasowy symulacji [s] | 0,1 | Wartości mnożników kryteriów funkcji celu | a=1  b=1000  c=0 |
| Prędkość wiatru (względem osi x i y) | x: 0  y: 0 |

*Wykres 6.5 Różnica pomiędzy najkrótszym a najdłuższym przelotem w symulacji*

Na wykresie 6.5 można odczytać średnią wartość funkcji celu wszystkich rozwiązań oraz wartości funkcji celu najlepszego i najgorszego rozwiązania. Jak można zauważyć w niektórych testach algorytm znalazł rozwiązania, w których różnica w czasach była nawet 3 razy mniejsza niż w najgorszym przypadku. Poza tym należy zauważyć, że najlepsze rozwiązania oscylowały między czterema a dziewięcioma sekundami. Przy wymiarach przestrzeni symulacji równych 100m x 100m x 100m oraz prędkości maksymalnej równej 5 takie wyniki są zadowalające. Algorytm w trakcie działania średnio odszukiwał rozwiązania lepsze od najgorszej wartości rozwiązania o 59%, zaś od średniej uzyskiwanej wartości o 37%.

## Maksymalizacja oddalenia dronów

Ostatnim badanym parametrem symulacji było oddalenie średnie dronów pomiędzy sobą. Kryterium to może być kluczowe jeśli chodzi o bezpieczeństwo, gdyż im większe oddalenie dronów, tym mniejsze prawdopodobieństwo kolizji. Może to być również pewien walor artystyczny, gdyż drony oddalone od siebie mogą wywierać na widzach większe wrażenie niż drony stłoczone w jednej części przestrzeni.

Wykonano testy na 10 zestawach pozycji początkowych i docelowych. Parametry symulacji pozostawały takie same (tabela 6.5). Mnożnik kryterium wynosił 2000, gdyż przy takich warunkach funkcja celu przyjmowała wartości zbliżone do czasu trwania symulacji.

*Tabela 6.5 Istotne parametry testowanych symulacji*

|  |  |  |  |
| --- | --- | --- | --- |
| Wymiary przestrzeni [m] | 100 x 100 x 100 | Siłą nośna drona [N] | 20 |
| Liczba dronów | 10 | Masa drona [kg] | 1 |
| Prędkość maksymalna | 5 | Użyty typ algorytmu | KLT+DLT |
| Interwał czasowy symulacji [s] | 0,1 | Wartości mnożników kryteriów funkcji celu | a=0  b=1000  c=2000 |
| Prędkość wiatru (względem osi x i y) | x: 0  y: 0 |

*Wykres 6.6 Średnie oddalenie pomiędzy dronami*

Zbiorem danych użytym do analizy były podobnie jak w przypadku poprzedniego testu wszystkie najlepsze wyniki w krokach algorytmu. Analizując wyniki widoczne na wykresie 6.6 można zauważyć, że różnice pomiędzy średnimi uzyskiwanymi wynikami, a najlepszymi rozwiązaniami nie jest zbyt duże. Mimo to średnie oddalenie pomiędzy dronami wynoszące ponad 50m w przestrzeni 100m x 100m x 100m jest dosyć zadowalające. Oczywiście kryterium to zmienia się w zależności od zestawu pozycji, co wykres pokazuje dosyć dobrze (różnica pomiędzy skrajnymi wartościami wynosi 25m). Z pewnością zwiększenie liczby dronów pogorszyłoby uzyskane wyniki.

# Podsumowanie

W niniejszej pracy inżynierskiej dokonano przeglądu informacji o dronach i ich wykorzystaniu oraz o algorytmach używanych do rozwiązywania problemu przydziału. W celu stworzenia symulatora lotu roju dronów w ograniczonej przestrzeni, opracowany został model matematyczny. Model uwzględnia przyspieszanie i hamowanie dronów, siły oporu aerodynamicznego (z uwzględnieniem ewentualnych wiatrów bocznych) i ciężkości oraz warunki techniczne drona. Opracowany został algorytm poszukiwań z zabronieniami, odpowiednio dostosowany do modelu i jego specyfiki, który za pomocą określonej funkcji celu dokonuje optymalizacji trajektorii dronów. Model matematyczny i algorytm zostały zaimplementowane w języku Python, a następnie przetestowane pod kątem prawidłowości obliczeń. Zrealizowane zostały testy sprawdzające zachowanie modelu i algorytmu dla różnych warunków i kryteriów. Podsumowując wykonane zostały wszystkie założenia dotyczącej tej pracy.

Wykonane testy potwierdzają skuteczność algorytmu w poszukiwaniu optymalnego (zgodnie z kryteriami) rozwiązania. W przypadku dążenia do maksymalizacji oddalenia dronów od siebie zaleca się użycie większej liczby dronów. Wyrównywanie czasu lotu wszystkich dronów pozwala na zmniejszenie różnicy czasu o ponad połowę w stosunku do najgorszych rozwiązań. Minimalna objętość przestrzeni przypadającą na jednego drona powinna wynosić (zgodnie z założeniami) 25m3. Prędkość maksymalna dronów nie powinna przekraczać 13.

W przyszłości można rozszerzyć stworzony model o trajektorie opierające się na pozycjach pośrednich. W rozwiązaniach, gdzie występują kolizje, możliwe byłoby dodanie pozycji pośredniej, która pozwalałaby na lot po trajektorii składającej się z dwóch odcinków leżących na prostych łączących tę pozycję z pozycją początkową i docelową. Dzięki temu, przy odpowiednich warunkach, możliwe byłoby wykorzystywanie również rozwiązań, które w obecnym modelu matematycznym są nieprzydatne. Dodatkowo można wprowadzić opóźnienie w rozpoczęciu lotu drona oraz zmianę wartości wykorzystywanej siły ciągu.

## 7. Odpis z bibliografii:

Rozdział 4:

[4.1] F.Glover, E. Taillard, D. de Werra: *A user's guide to tabu search.* Annals of Operations Research 41: 3-28, 1993r.

(użyte w rozdziale 2)[4.2] K. Wala: *Algorytm Tabu w optymalizacji uogólnionego problemu przydziału.* Automatyka Tom 11 Zeszyt 1-2: 309-316, 2007r.

Rozdział 2:

[2.1] S. Dasgupta, C. Papadimitriou, U. Vazirani: *Algorytmy* Wydawnictwo Naukowe PWN, 2012r.

[2.2] *Algorytmy heurystyczne*, [Online]: <http://algorytmy.ency.pl/artykul/algorytmy_heurystyczne>

[2.3] J. Pempera *Równoległy algorytm tabu z elementami inspirowanymi naturą dla problemu planowania tras.* Automatyka Tom 15 Zeszyt 2: 399-408, 2011r.

[2.4] J. Grabowski, J. Pempera *Hybrydowy algorytm tabu dla niepermutacyjnego problemu przepływowego z kryterium sumacyjnym.* Automatyka Tom 13 Zeszyt 2: 289-296, 2009r

[2.5] R. Kowalik, T. Łusiak, A. Novak *A mathematical model for controlling a quadrotor UAV.* Transactions on Aerospace Research Rocznik 2021 Tom 3 (264): 58-70, 2021r.

[2.6] D. Tobór, J. Barcik, P. Czech *Legal aspects of air transport safety and the use of drones*. Scientific Journal of Silesian University of Technology. Series Transport Tom 97: 167-179, 2017r.

[2.7] A. Miah *Drones : The Brilliant, the Bad and the Beautiful.* Wydawnictwo Emerald publishing, 2020r.

[2.8] M. Feltynowski, M. Zawistowski *Możliwości wykorzystania bezzałogowych platform w służbach ratunkowo-porządkowych.* Bezpieczeństwo i technika pożarnicza Tom 51: 138-149, 2018r.

[2.9] N. Schwartz *Watch 1,218 drones form the Olympics rings during Opening Ceremony* [online] <https://ftw.usatoday.com/2018/02/2018-winter-olympics-opening-ceremony-1218-drones-rings>

[1] P. Bukowski, G. Szala *Bezzałogowe statki powietrzne – geneza, teraźniejszość i przyszłość* Postępy w Inżynierii Mechanicznej Tom 11(6): 5-11, 2018r.

# [2] MRT *How many thousands of drones have they used in the balloon of the Tokyo 2020 ceremony?* [online] <https://marketresearchtelecast.com/how-many-thousands-of-drones-have-they-used-in-the-balloon-of-the-tokyo-2020-ceremony/111062/>

# [15] Damien A. *Spyder, potężne interaktywne środowisko programistyczne dla języka Python* [online] <https://ubunlog.com/pl/spyder-entorno-desarrollo-python/>