



Politechnika Wrocławska

Wydział Informatyki i Zarządzania  
Kierunek studiów: Informatyka

## Praca dyplomowa – magisterska

# **TWORZENIE ZACHOWAŃ STADNYCH PRZEZ WIRTUALNYCH AGENTÓW PRZY ZASTOSOWANIU GŁĘBOKICH SIECI NEURONOWYCH**

Adrian Ślimak

słowa kluczowe:  
wirtualne środowisko, sztuczne życie, głębokie  
sieci neuronowe, zachowania stadne

### krótkie streszczenie:

Praca skupia się na symulacji sztucznego życia, w którym w postaci agentów występują ofiary i drapieżniki. Ich sterowniki bazują na sztucznych sieciach neuronowych - RNN i LSTM. Celem pracy była ocena obu podejść, jak również zbadanie wpływu komunikacji w obrębie danego gatunku na przeżywalność agentów.

Opiekun pracy dyplomowej	.....	.....
Tytuł/stopień naukowy/imię i nazwisko	ocena	podpis
Ostateczna ocena za pracę dyplomową		
Przewodniczący Komisji egzaminu dyplomowego	.....	.....
Tytuł/stopień naukowy/imię i nazwisko	ocena	podpis

*Do celów archiwalnych pracę dyplomową zakwalifikowano do:\**

- a) kategorii A (akta wieczyste)
- b) kategorii BE 50 (po 50 latach podlegające ekspertyzie)

\* niepotrzebne skreślić

pieczętka wydziałowa

Wrocław, rok 2020



## **STRESZCZENIE**

Przegląd literatury w dziedzinie ewolucji zachowań stadnych sztucznego życia wykazał, że istniejące badania koncentrują się głównie wokół presji wywieranej przez obecność drapieżnika oraz tworzeniu stad przez ofiary w odpowiedzi na różne formy ataków. Brak natomiast badań, które szerzej poruszyłyby kwestie tak kluczowe w naturze, jak konieczność zdobywania pożywienia przez ofiary, czy komunikacja wśród drapieżników, która mogłaby umożliwić przełamanie formy obrony ofiar jaką jest stado.

Niniejsza praca wypełnia tę lukę. W tym celu skonstruowano wirtualne środowisko, które zawiera niezbędne mechanizmy oraz modeluje wybrany fragment natury w stopniu wymaganym do przeprowadzenia badań. Środowisko zamieszkuje agenty, czyli sztuczne zwierzęta, których zachowanie jest określone za pomocą sterownika zbudowanego na bazie rekurencyjnej sieci neuronowej RNN albo sieci LSTM. Na podstawie obserwacji odbieranych ze środowiska za pomocą zmysłów oraz własnego stanu pamięci, sterownik decyduje o zmianie położenia agenta lub o wydaniu dźwięku.

W pracy przeprowadzono porównanie rodzaju zastosowanego sterownika w badaniach nad tworzeniem się zachowań stadnych wśród agentów, żyjących pod presją ataków drapieżnika i konieczności poszukiwania pożywienia, a także wpływu mechanizmu komunikacji obecnego zarówno u ofiar jak i drapieżników.

## **ABSTRACT**

A review of the literature on the evolution of artificial life herd behaviour has shown that existing research focuses mainly on the pressure exerted by the presence of a predator and the creation of herds by prey in response to different forms of attacks. However, there is a lack of research that would address more broadly such key issues in nature as the need for prey to feed or communication among predators, that could allow breaking through the form of prey defense that is a herd.

This work fills this gap. For this purpose, a virtual environment has been constructed, which contains the necessary mechanisms and models a selected fragment of nature to the extent required for research. The environment is inhabited by agents, that is, artificial animals, which are controlled by a driver based on a recurrent neural network RNN or LSTM. Based on the observations received from the environment by the senses and their memory state, the driver decides to change the position of the agent or to release a sound.

In this thesis, a comparison was made of the type of controller used in the research on the formation of herd behaviour among agents, living under the pressure of predator attacks and the need to search for food, as well as the impact of the communication mechanism present in both preys and predators.



# SPIS TREŚCI

<b>Streszczenie . . . . .</b>	1
<b>Wstęp . . . . .</b>	4
<b>1. Przegląd literatury . . . . .</b>	6
1.1. Presja ataku drapieżników . . . . .	6
1.2. Mechanizm komunikacji . . . . .	9
1.3. Algorytm stadny . . . . .	10
1.4. Bliżej rzeczywistości . . . . .	11
1.5. Podejście matematyczne . . . . .	12
1.6. Podsumowanie . . . . .	13
<b>2. Wirtualny świat . . . . .</b>	14
2.1. Koncepcja sztucznego życia . . . . .	14
2.2. Środowisko badawcze . . . . .	15
<b>3. Zastosowane modele . . . . .</b>	17
3.1. Sztuczna sieć neuronowa . . . . .	17
3.2. Rekurencyjna sieć neuronowa . . . . .	19
3.3. Sieć rekurencyjna LSTM . . . . .	20
3.4. Algorytm genetyczny . . . . .	22
<b>4. Środowisko badawcze . . . . .</b>	25
4.1. Agenty . . . . .	26
4.1.1. Agent ofiary . . . . .	26
4.1.2. Agent drapieżnika . . . . .	26
4.1.3. Zmysły agentów . . . . .	27
4.1.4. Sterownik agentów . . . . .	29
4.2. Algorytm genetyczny . . . . .	30
4.2.1. Genotyp . . . . .	30
4.2.2. Operatory ewolucyjne . . . . .	32
4.2.3. Funkcja przystosowania . . . . .	33
4.3. Symulacja . . . . .	33
<b>5. Badania eksperymentalne . . . . .</b>	37
5.1. Seria I - Strojenie parametrów algorytmu genetycznego . . . . .	38
5.1.1. Badania wpływu parametrów algorytmu genetycznego dla sterownika będącego siecią RNN . . . . .	38
5.1.2. Badania wpływu parametrów algorytmu genetycznego dla sterownika będącego siecią LSTM . . . . .	48
5.1.3. Podsumowanie badań - wybór wartości parametrów algorytmu genetycznego . . . . .	57
5.2. Seria II - Badanie wpływu liczby neuronów ukrytych sterownika ofiar na tworzenie się zachowań stadnych przy presji ataków drapieżnika . . . . .	58
5.2.1. Badania dla sterownika wykorzystującego sieć RNN . . . . .	59
5.2.2. Badania dla sterownika wykorzystującego sieć LSTM . . . . .	63
5.2.3. Porównanie wyników eksperymentów serii II . . . . .	67

5.3.	Seria III - Wpływ mechanizmu komunikacji ofiar na tworzenie się zachowań stadnych przy presji spowodowanej obecnością wielu drapieżników . . . . .	69
5.3.1.	Badania dla sterownika wykorzystującego sieć RNN . . . . .	70
5.3.2.	Badania dla sterownika wykorzystującego sieć LSTM . . . . .	74
5.3.3.	Porównanie wyników eksperymentów serii III . . . . .	78
5.4.	Seria IV - Wpływ liczby drapieżników obecnych w środowisku na skuteczność przeprowadzanych przez nie ataków . . . . .	80
5.4.1.	Badania dla sterownika wykorzystującego sieć RNN . . . . .	81
5.4.2.	Badania dla sterownika wykorzystującego sieć LSTM . . . . .	85
5.5.	Seria V - Wpływ sposobu rozmieszczenia pożywienia w środowisku na formowanie się zachowań stadnych u ofiar . . . . .	88
5.5.1.	Badania dla sterownika wykorzystującego sieć RNN . . . . .	89
5.5.2.	Badania dla sterownika wykorzystującego sieć LSTM . . . . .	96
5.6.	Seria VI - Bezpośrednie porównanie sterownika bazującego na sieci RNN oraz LSTM .	102
<b>6.</b>	<b>Podsumowanie badań eksperimentalnych . . . . .</b>	<b>108</b>
<b>7.</b>	<b>Zakończenie . . . . .</b>	<b>110</b>
<b>Bibliografia . . . . .</b>		<b>112</b>
<b>Spis rysunków . . . . .</b>		<b>113</b>
<b>Spis tabel . . . . .</b>		<b>118</b>
<b>Dodatki . . . . .</b>		<b>119</b>
Zawartość płyty DVD . . . . .		119
Wizualizacje . . . . .		119

## **WSTĘP**

Badania nad zwierzętami, a w szczególności obserwacje ich życia w naturalnym środowisku, są prowadzone od wieków. U wielu gatunków występują złożone zachowania kolektywne [9, 20]. Delfiny na przykład, wykształciły zaawansowane metody komunikacji składające się z gestów i sygnałów dźwiękowych, a pszczoły potrafią przekazywać informacje, wykorzystując w tym celu taniec. Szympansy i wilki żyją w rozbudowanych hierarchiach społecznych, natomiast mrówki, wspólnymi siłami potrafią dokonywać rzeczy, których pojedynczo nigdy nie byłyby w stanie osiągnąć. Analizując zachowanie przedstawicieli różnych gatunków, naukowcy dokonali fascynujących odkryć, które znalazły zastosowanie w wielu dziedzinach nauki i techniki.

Grupowanie się zwierząt jest zjawiskiem powszechnie występującym w naturze i niesie ze sobą wiele korzyści. Dla gatunków należących do grupy ofiar, kluczowe jest zapewnienie bezpieczeństwa przed drapieżnikami. Duża liczba osobników poruszająca się w stadzie, może wywołać efekt dezorientacji wśród drapieżników, co znacznie zmniejsza ich zdolność skupienia się i śledzenia pojedynczego celu, redukując skuteczność ataków.

Gatunki drapieżne wykształciły zróżnicowane techniki polowania, począwszy od obierania za cel osłabione lub samotne jednostki, po bardziej zaawansowane strategie zakładające grupowe ataki, w których każdy osobnik ma własne zadanie, np. początkowe rozproszenie stada ofiar tak, aby ułatwić innym polowanie na odizolowane, oddalające się od stada ofiary. Takie podejście znacznie podnosi szansę powodzenia i zdobycia pożywienia, niż samotne łowy.

Zwierzęta w stadach również zyskują dzięki tak zwanemu efektowi "wielu oczu". Im większa liczba osobników, tym większa szansa, że któryś zdoła odpowiednio wcześnie dostrzec zbliżającego się drapieżnika i ostrzec pozostałych przed niebezpieczeństwem. Taka zbiorowa czujność niesie również inne korzyści. Pojedyncze jednostki mogą poświęcić więcej czasu i energii na poszukiwanie pożywienia, a wspólnymi siłami przeczesać znacznie większy obszar.

Korzyści nie kończą się jednak na unikaniu drapieżników. Zwierzęta, które żyją w grupach, nie muszą wędrować po okolicy w poszukiwaniu odpowiedniego partnera, co jest problemem dla gatunków samotnie żyjących, zamieszkujących duże obszary. Kiedy na świecie pojawi się już potomstwo, dorosłe osobniki mogą współpracować w celu ochrony i zapewnienia pożywienia młodemu pokoleniu. Istnieją również inne, nieco mniej oczywiste korzyści z życia grupowego. Wspólnie latające ptaki lub ryby płynące w ławicy, poruszają się znacznie sprawniej niż samotnicy, dzięki czemu oszczędzają energię niezbędną podczas długich migracji. Zwiększa się również szansa przeżycia w trudnych warunkach środowiskowych, jak w przypadku nietoperzy oraz pingwinów, które wzajemnie się ogrzewają.

Przy tak wielu korzyściach, może wydawać się zaskakujące, że istnieją gatunki unikające swoich pobratymców, lecz w niektórych przypadkach, dzielenie się jedzeniem nie jest warte zalet płynących z życia w grupie. Kluczowym czynnikiem okazuje się zapewnienie wystarczającej ilości pożywienia, co może okazać się niewykonalne przy dużym apetycie.

Zachowania stadne zwierząt są tematem szerokich badań, głównie przez wzgląd na potencjalne zastosowania w inżynierii, robotyce czy sztucznej inteligencji, a także korzystny wpływ na zrozumienie inteligencji społecznej [23]. Jednym z ważniejszych aspektów tych badań są próby odkrycia czynników sprzyjających ewolucji takich zachowań. Jest to wymagające zadanie, ponieważ obserwacje zwierząt w ich naturalnym środowisku są trudne i często niebezpieczne, a

przede wszystkim czasochłonne. Większość gatunków potrzebuje miesięcy, a nawet lat, żeby wydać na świat potomstwo, a zauważalne zmiany mogą być widoczne dopiero za kilka pokoleń.

Z pomocą przychodzi technologia, pozwalająca na stworzenie wirtualnych środowisk, które w znacznym uproszczeniu, umożliwiają symulację warunków panujących w przyrodzie. Takie podejście pozwala pozbyć się problemu długotrwałych obserwacji zwierząt, kolejne pokolenia pojawiają się szybciej, a ponadto daje możliwość badania wyłącznie wybranych czynników. Tego typu symulacje mogą pomóc w udzieleniu odpowiedzi na pytania dotyczące ewolucji, świadomości i zachowania zwierząt, a nawet przybliżyć nas do stworzenia prawdziwej sztucznej inteligencji.

Natura cieszy się tak bogatą różnorodnością wszelkich form życia, w głównej mierze dzięki ewolucji. Jest to powolny proces stopniowych zmian dziedziczonych cech, zachodzący na przestrzeni kolejnych pokoleń. Ewolucja umożliwia organizmowi przystosowanie się do życia w określonym środowisku, a także adaptację do nieustannie zmieniających się warunków.

Sam proces ewolucji stanowił inspirację do stworzenia algorytmów ewolucyjnych, które z powodzeniem wykorzystywane są w wielu problemach optymalizacyjnych. W związku z tym, doskonale nadają się również do symulacji zjawisk ewolucyjnych w sztucznych środowiskach. Agenty, czyli wirtualni przedstawiciele organizmów, mogą ewoluować na kilka sposobów. Naturalnym rozwiązaniem, jest samodzielne poszukiwanie partnerów i wydawanie potomstwa, lecz bardziej praktycznym i częściej stosowanym podejściem jest sztuczne rozmnażanie, wykorzystując mechanizmy selekcji i krzyżowania oraz okazyjne mutacje.

W rzeczywistości, ewolucji podlegają wszystkie cechy organizmu. W symulacji natomiast, ze względu na stosowane uproszczenia, ewolucja dotyczy zazwyczaj struktury sztucznego mózgu (sterownika) agentów. Polega ona na modyfikacji parametrów zastosowanego sterownika, np. wag w przypadku sieci neuronowych, co bezpośrednio wpływa na zmianę zachowania. Zdarzają się również symulacje, w których ewolucji podlegają zdolności, czy nawet elementy budowy ciała osobników.

Celem niniejszej pracy jest zbadanie użyteczności sieci neuronowej LSTM jako sterownika agenta ofiary i drapieżnika oraz porównanie jej skuteczności ze sterownikiem będącym klasyczną siecią rekurencyjną (RNN). Ponadto zbadany zostanie wpływ komunikacji dźwiękowej i użytych sterowników na tworzenie się zachowań stadnych, a także innych strategii umożliwiających przetrwanie w wirtualnym środowisku.

Realizacja celu pracy wymagała opracowania wirtualnego środowiska oraz implementacji mechanizmu komunikacji agentów, mechanizmu dezorientacji drapieżników oraz mechanizmu zdobywania pożywienia i energii życiowej ofiar. Zaimplementowano i przebadano dwa rodzaje sterownika agentów, bazujących na rekurencyjnej sieci RNN oraz LSTM. Umożliwiło to przeprowadzenie badań mających na celu porównanie skuteczności obu sterowników oraz badań nad wpływem komunikacji dźwiękowej na tworzenie się zachowań stadnych.

Praca składa się z siedmiu rozdziałów. W rozdziale 1 zamieszczono przegląd i analizę stanu literatury na temat podjętego problemu. Rozdział 2 poświęcono na przedstawienie koncepcji sztucznego życia oraz idei tworzenia wirtualnych środowisk. W rozdziale 3 zawarto teoretyczny opis wykorzystanych modeli sieci neuronowych oraz algorytmu genetycznego. Rozdział 4 przedstawia opis przygotowanego środowiska oraz przebiegu symulacji, a także omówienie budowy agentów i sterowników. W rozdziale 5 przeprowadzone zostały badania, które umożliwiły porównanie sterowników oraz wpływu poszczególnych mechanizmów środowiska na zachowanie agentów. Na koniec, w rozdziale 6 przedstawiono analizę i podsumowanie uzyskanych wyników, a w rozdziale 7 podsumowano pracę oraz zaproponowano kierunki dalszego rozwoju.

# **1. PRZEGŁĄD LITERATURY**

Niniejszy rozdział przedstawia przegląd literatury dotyczącej problemu formowania się zachowań stadnych u agentów w wirtualnych środowiskach.

Badania nad zwierzętami są powszechnne, ale w rzeczywistości trudno znaleźć odpowiedzi co do przyczyn stojących u podstaw ewolucji obserwowanych zachowań. Ze względu na złożoność natury i mnogość czynników wzajemnie oddziałujących na siebie, nie sposób określić ich znaczenie. W tym celu powstają prace, które wykorzystując możliwości obliczeniowe komputerów, tworzą wirtualne światy, które dzięki pełnej kontroli nad warunkami w nich panującymi, pozwalają na badania wybranych aspektów życia zwierząt.

Prac zajmujących się badaniem problemu sztucznego życia i ewolucji zachowań stadnych wciąż nie ma wiele. Większość z nich skupia się na określonych scenariuszach i badaniu wybranych aspektów środowiska naturalnego, w mocno uproszczonych warunkach symulacyjnych. Prace koncentrują się głównie na ewolucji zmysłów i sztucznego "mózgu" zwierząt, natomiast szczegółowe dotyczące budowy ich ciał są całkowicie pomijane. Różnią się natomiast podejściem, przyjętymi założeniami i ograniczeniami, czy wykorzystanym sterownikiem agentów. Istnieją również prace, które starają się w sposób formalny przedstawić rozwiązanie omawianego problemu, w podrozdziale 1.5 znajduje się opis jednej z nich.

## **1.1. PRESJA ATAKU DRAPIEŻNIKÓW**

Jedna z teorii mówi, że przyczyną grupowania się zwierząt jest efekt dezorientacji wywoływanego pośród atakujących je drapieżników, który wynika z obecności wielu, szybko poruszających się osobników stada. Przeprowadzono w tym kierunku wiele obserwacji oraz badań, między innymi z wykorzystaniem Dafni oraz gatunku niewielkich ryb (Cierników), gdzie wykazano, że przy większym zagęszczeniu grupy Dafni, skuteczność przeprowadzanych wobec nich ataków istotnie zmalała [10]. Wciąż jednak nie do końca jest jasne, jak ważna jest ta selektywna siła.

W pracy [15], powyższe twierdzenie poddane zostało weryfikacji przy wykorzystaniu wirtualnego środowiska. Na potrzeby przeprowadzonych w niej badań, stworzono symulację, w której ofiary i drapieżnicy zamieszkują dwuwymiarową przestrzeń. Agenty wyposażono jedynie w zmysł wzroku, którego pole podzielono na równe segmenty. Stanowi on źródło informacji docierającej do mózgu i potrafi określić rodzaj stworzenia, a także w którym segmencie obszaru wzroku się ono znajduje. Każdy agent jest sterowany przez sieć Markowa, która na podstawie sensorycznego wejścia (zmysłu wzroku) oraz stanów wewnętrznych (pamięci), podejmuje decyzję dotyczącą kolejnej akcji. Organizm może wykonać ruch w przód, w lewo i prawo lub pozostać w miejscu. Niektóre cechy agentów, takie jak prędkość ruchu i skrętu, zasięg i kąt pola widzenia czy długość przerwy pomiędzy kolejnymi atakami drapieżnika, są stałe i mają zamodelować różnice między gatunkami zwierząt występujące w przyrodzie. Mechanizm dezorientacji drapieżników został zaimplementowany jako ograniczenie w ich percepceji (dokładny sposób jego działania omówiono w rozdziale 4). W trakcie symulacji, do środowiska trafiają przedstawiciele obu gatunków. Każdy z nich posiada własny genotyp, który opisuje strukturę jego sterownika. Agenty podczas symulacji są oceniane pod względem przystosowania do życia w środowisku, a proces ewolucji realizowany jest z wykorzystaniem algorytmu genetycznego.

W przeprowadzonych badaniach udało się zaobserwować znaczne różnice w zachowaniu ofiar. Kiedy drapieżnik mógł polować swobodnie, bez ograniczeń, atakując wybierał najbliższy cel, natomiast agenci ofiar starały się rozproszyć najbardziej jak to było możliwe. W przypadku, kiedy ewolucja następowała przy obecnym efekcie dezorientacji, ofiary wykazywały zachowania stadne i dążyły do grupowania się w ponad 70% z przeprowadzonych symulacji. Ataki drapieżnika na grupę są mniej skuteczne, dlatego wykształcił strategię polowania na osobniki znajdujące się na skraju stada.

Dodatkowo, sprawdzono jak na efekt dezorientacji wpływa kąt pola widzenia drapieżnika. Przy  $60^\circ$ , nie miał on istotnego znaczenia na skuteczność ataków, więc grupowanie się ofiar przestało być dobrą strategią obronną. Kiedy pole widzenia było większe niż  $120^\circ$ , dezorientacja znów był na tyle silna, że przyczyniła się do formowania stad.

Udało się wykazać, że efekt dezorientacji prawdopodobnie przyczynił się do ewolucji zachowań stadnych u zwierząt, na które polowały drapieżniki wykorzystujące zmysł wzroku do śledzenia swojej ofiary.

Praca [1] stanowi kontynuację badań opisanych powyżej. Tym razem, autorzy zdecydowali się sprawdzić popularne w biologii powiedzenie: "Oczy z przodu, zwierzę poluje. Oczy z boku, zwierzę się ukrywa.". Powszechnie uważa się, że system wzrokowy o dużej ostrości, skierowany w przód, jaki obserwuje się u wielu gatunków drapieżniczych, odgrywa ważną rolę podczas polowania, brak jednak dowodów empirycznych, które wyjaśniały historię jego ewolucji.

W tym artykule, próbie poddana została hipoteza mówiąca, że dobrze znany efekt dezorientacji drapieżnika, mógł doprowadzić do ewolucji złożonego systemu wzrokowego. Na potrzeby pracy wykorzystano środowisko symulacyjne opracowane poprzednio, lecz dodatkowo, pozwolono na ewolucję zmysłu wzroku drapieżników, co umożliwia zbadanie sposobu w jaki dostosowuje się on do zachowania agentów ofiar.

Przeprowadzone obserwacje pokazały, że taka swoboda ewolucji prowadzi do sytuacji, w której oba gatunki naprzemiennie dostosowują swoje cechy. Można również wykazać negatywną korelację między kątem widzenia, a średnią gęstością stada ofiar.

To zjawisko przedstawia następujący cykl:

1. W odpowiedzi na ataki drapieżników, ofiary ewoluują mechanizm obronny polegający na łączeniu się w stada.
2. Drapieżcy zawężają swoje pole widzenia, aby zminimalizować wpływ efektu dezorientacji i skuteczniej polować na grupy ofiar.
3. Ofiary porzucają już nieskuteczny mechanizm grupowania się i zwiększają swoje rozproszenie.
4. Pole widzenia drapieżników ponownie rozszerza się i cykl wraca do punktu 1.

Odkrycie to jest zaskakujące, ponieważ populacja drapieżników powinna skutecznie "pokonać" grupujące się ofiary zawężając wzrok. Dlaczego więc decydują się poszerzyć swoje pole widzenia, gdy ofiary się rozproszą i pozwolić na ponowną ewolucję zachowań stadnych? Okazuje się, że drapieżnik z niewielkim kątem wzroku, mając do czynienia z rozproszoną zdobyczą, ma mniejsze szanse naauważenie ofiary, w związku z czym selekcja ponownie faworyzuje osobniki o dużym polu widzenia.

Autorom udało się pokazać, że sprzeczne ze sobą wymagania systemu wzrokowego drapieżnika, wywołane efektem dezorientacji, mogą być przyczyną ewolucji złożonych oczu, które zaspokajają obie potrzeby. Szerszy, mniej skupiony wzrok jest bardziej użyteczny przy początkowym wykrywaniu ofiar, natomiast skoncentrowany, o wysokiej ostrości, lepiej sprawdza się do śledzenia pojedynczego osobnika i unikania skutków efektu dezorientacji podczas polowania na grupy zwierząt.

Praca [16] jest kontynuacją poprzednio opisanych artykułów. Do symulacji wykorzystano to samo środowisko, lecz agenty drapieżników zostały wyłączone z procesu ewolucji, a ich zachowanie modeluje jeden z zaimplementowanych algorytmów. Takie podejście zostało zastosowane, aby móc zbadać wpływ zróżnicowanych strategii ataków na zachowanie się populacji ofiar. Przeprowadzone badania dotyczą czterech strategii ataków:

1. Ataki losowe (ang. *Random Attacks*).

Atak odbywa się poprzez wybranie losowego punktu w przestrzeni symulacji, a następnie zlikwidowanie ofiary znajdującej się najbliżej. To wstępne badanie miało na celu weryfikację hipotezy Hamiltona dotyczącej samolubnego stada, poprzez zamodelowanie sytuacji w której ofiary atakowane są przez ukrytego drapieżnika i nie wiedzą skąd nastąpi atak. Jeśli hipoteza jest słuszna, oczekuje się, że każdy agent dąży do minimalizacji swojej strefy zagrożenia (ang. *domain of danger*) i zwiększenia szansy na przeżycie, poprzez chowanie się wśród osobników swojego gatunku.

Obserwacje wykazały, że losowe ataki są wystarczające do wykształcenia się mechanizmu grupowania ofiar, co potwierdza postawioną hipotezę. Czas potrzebny na ewolucję grupowania się był około pięciokrotnie wyższy niż w przypadku dwóch kolejnych rodzajów ataków.

2. Losowe ataki wędrujące (ang. *Random Walk Attacks*)

Pierwszy atak odbywa się losowo, natomiast każdy kolejny następuje w odległości nie większej niż 50 wirtualnych metrów od miejsca poprzedniego ataku. Ma to modelować drapieżnika nieustannie żerującego na grupie ofiar.

Tego typu atak wymusił na ofiarach znacznie szybszą ewolucję zachowań stadnych niż w poprzednim przypadku, a ponadto formujące się grupy były znacznie ścisłejsze.

3. Ataki z zewnątrz (ang. *Outside Attacks*)

Atak polega na losowym wyborze kąta, który jest przekształcany na odpowiadający mu punkt z krawędzi środowiska. Następnie, zaatakowana zostaje ofiara znajdująca się najbliżej tego miejsca. Odzwierciedla to drapieżnika nadchodzącego od zewnątrz i atakującego ofiary znajdujące się na skraju stada lub oddalające się od niego.

Ta strategia wywarła największą presję na ofiarach i spowodowała najszybsze wykształcenie się mechanizmu grupowania, a powstałe stada były najścisłejsze.

4. Ataki obszarowe miejsc o dużej gęstości (ang. *High-Density Area Attacks*)

Poprzednie ataki skutkowały wykształceniem się formy obrony w postaci zachowań stadnych. Istnieją również strategie obserwowane wśród naturalnych drapieżników, które są odpowiedzią na grupujące się ofiary. Przykładem są półwale błękitne, których obszarowe ataki koncentrują się na najgęstszych miejscach ławic Kryli, konsumując setki osobników jednocześnie. Kryle doświadczają również pojedynczych ataków ze strony mniejszych drapieżników, więc pojawia się pytanie, dlaczego grupują się w ogromne ławice, skoro ataki półwali przynoszą im tak duże straty?

W celu zbadania wpływu ataków obszarowych na formowanie się zachowań stadnych, zamodelowano sytuację, w której ofiary doświadczały dwa typy ataków (obszarowych oraz z zewnątrz). Przeprowadzono dwa testy, w których ataki obszarowe występowały odpowiednio 2,5 i 25 razy rzadziej niż ataki pojedyncze. Pierwszy przypadek pokazał, że presja ze strony ataków obszarowych była zbyt silna, aby pozwolić na wykształcenie się zachowań stadnych. W drugim przypadku udało się je zaobserwować, lecz nastąpiło to wyraźnie później niż przy wyłącznych atakach z zewnątrz.

Badaniami udało się pokazać, że ataki anty-stadne, celujące w obszary o największej gęstości grupy, spowalniają wykształcenie się mechanizmów grupowania, lecz o ile nie następują zbyt często, nie powstrzymują ich, co wyjaśnia zachowanie ławic Kryli.

## **1.2. MECHANIZM KOMUNIKACJI**

W środowisku naturalnym, zdolność komunikacji między przedstawicielami poszczególnych gatunków, zarówno wśród ofiar jak i drapieżników, odgrywa bardzo ważną rolę. Może ona odbywać się, na przykład drogą zapachów pozostawianych w środowisku, jak wśród mrówek, czy skomplikowanego tańca u pszczół. Najczęściej jednak, komunikacja odbywa się przy pomocy zmysłu słuchu oraz wydawanych dźwięków. Zaletą takiego rozwiązania jest możliwość przekazywania informacji na dalekie odległości oraz bez konieczności pozostawania w zasięgu wzroku. Umiejętność wymiany informacji ma istotny wpływ na zdolność przetrwania w środowisku. Zwierzęta żyjące w stadach mogą komunikować się w celu zdobycia pożywienia, przeszukując tym samym znacznie większy obszar, ale przede wszystkim, zapewnia ona możliwość wczesnego ostrzeżenia pozostałych członków grupy przez zbliżającym się niebezpieczeństwem, jeszcze zanim zostanie dostrzeżone przez wszystkich. Drapieżnicy z kolei mogą wykorzystać umiejętności komunikowania się przy bardziej złożonych strategiach polowania, ale także wykorzystać dźwięki wydawane przez ofiary do lokalizowania ich, w tym przypadku mechanizm komunikacji zadziała przeciwko nim.

Mimo dużego znaczenia w świecie przyrody, mechanizm komunikacji nie jest popularnym kierunkiem badań. W pracy [12], postanowiono to zmienić i przeprowadzono analizę wpływu mechanizmu komunikacji na kształtowanie się zachowań stadnych u zwierząt. Na potrzeby badań, które stanowią rozszerzenie tych przedstawionych w artykule [1] i [16], wykorzystano takie samo środowisko, co pozwoliło porównać otrzymane wyniki. Jako sterownik agentów, oprócz sieci Markowa, zastosowano znacznie bardziej złożoną, klasyczną rekurencyjną sieć neuronową (opisaną dokładniej w rozdziale 3.2). Zdolność komunikacji została zapewniona jedynie agentom ofiar. Posiadają one zmysł słuchu, który wykrywa kierunek dźwięków wydawanych przez pobliskie osoby, ograniczonych do jednego rodzaju.

Przeprowadzone badania wykazały, że umiejętność komunikacji ofiar, skutkowała zmniejszoną potrzebą życia w ścisłym stanie. Agenty wykształcili zdolność ostrzegania, pozwalającą odpowiednio wcześnie informować oraz reagować na zagrożenie. Dzięki temu, mimo znacznie niższej gęstości stada oraz wyższej jego rozpiętości, były w stanie osiągnąć porównywalną przeżywalność, co do osobników niewyposażonych w zdolność komunikacji. Zaobserwowano również zmniejszoną liczbę prób ataków drapieżnika.

Powyzsze wyniki odnoszą się do sterownika będącego rekurencyjną siecią neuronową. W przypadku sieci markowej, nie udało się zaobserwować znacznego wpływu mechanizmu komunikacji na zachowanie agentów, co może wynikać z prostej struktury budowy tego sterownika, który nie był w stanie nauczyć się wykorzystania dodatkowych możliwości oferowanych przez komunikację. Ponadto, sieci neuronowe zapewniły szybszy proces uczenia się, wykazały większą stabilność oraz przede wszystkim, wyższe wartości przystosowania agentów. Według autora, przyczyna takiego stanu rzeczy może tkwić w stochastycznym charakterze sieci Markowa, która mimo wyuczenia się, ma szanse wygenerować decyzje mało prawdopodobną, która może okazać się błędna i przyczynić do śmierci ofiary.

### 1.3. ALGORYTM STADNY

Przeważnie jako sterownik agenta wykorzystuje się model bazujący na sieci Markowa [1, 15, 16] lub sieci neuronowej [3, 21], który na podstawie wejścia obserwacji względnie sam podejmuje decyzje. W przypadku prac [5, 24], omówionych niżej, zachowanie każdego z osobników jest zdeterminowane przez algorytm, natomiast ewolucji podlegają jego parametry, cechujące daną jednostkę.

Takie rozwiązanie oparto na pracy Craiga Reynolds [17], który w 1986 roku, przedstawił komputerowy model skoordynowanego ruchu zwierząt, z dużym podobieństwem imitującym zachowanie stada ptaków czy ławicy ryb. Podstawowy algorytm działa na zasadzie trzech prostych reguł sterowania, określających zachowanie pojedynczych osobników, nazywanych również boidami (ang. *Boids*). Każda reguła posiada odpowiadające jej pole: separacji, wyrównania oraz spójności, o określonym zasięgu i kącie, które wykrywają bliskie stworzenia. Na podstawie względnej pozycji i wektora prędkości obiektów znajdujących się w obszarze każdego z pól, wyliczany jest wypadkowy kierunek i prędkość poruszania się danego agenta.

Jedną z prac wykorzystujących takie podejście jest [24]. Postanowiono w niej zbadać jak na ewolucję zachowań stadnych wpływa obecność drapieżnika, a dodatkowo wprowadzono element poszukiwania pożywienia przez ofiary, który był badany niezależnie od występującej presji drapieżników. Agenty były nagradzane większą wartością funkcji przystosowania za zdobycie pokarmu, lecz nie mogły zginąć z powodu niedostatecznej jego ilości. Podczas symulacji, ważnym elementem, na który autorzy zwróciли uwagę, jest faza rozgrzewki, która musi nastąpić zanim w środowisku pojawi się drapieżnik lub pożywienie.

Każdy z agentów posiada zestaw cech określających właściwości takie jak zasięgi pól, kąt widzenia, prędkość ruchu i skrętu, a także siła odpychająca od drapieżników i przyciągająca do pożywienia. Wszystkie parametry podlegają ewolucji, lecz dla części z nich zdefiniowano przedziały dopuszczalnych wartości oraz ograniczenia mające zapobiec powstawaniu agentów, maksymalizujących wszystkie możliwe cechy. Zamiast tego, ewolucja powinna znaleźć rozwiązanie zapewniające równowagę pomiędzy kontrastującymi właściwościami. Oznacza to, że daleki zasięg wzroku skutkuje mniejszym kątem widzenia, natomiast wysoka prędkość poruszania się ogranicza zwinnosć, czyli szybkość skrętu.

W przeprowadzonych badaniach zauważono, że presja drapieżnika prowadziła do formowania się dwóch rodzajów stad. Pierwsze z nich ewoluowało zgodnie z zasadą "samolubnego stada", w której każdy osobnik próbował minimalizować swoją strefę zagrożenia, kierując się do środka grupy i ukrywając pośród innych organizmów. Drugi rodzaj charakteryzował się wysoką dynamiką i szybko poruszającymi się agentami. Podczas ataku drapieżnika, stado szybko rozdzielało się na dwie mniejsze grupy, omijało niebezpieczeństwo, a następnie powoli znów łączyło w całość. Zaobserwowano również, że ofiary z niewielkim kątem widzenia były łatwo eliminowane przez drapieżniki, co skutkowało ewolucją organizmów o dużym polu widzenia wynoszącym powyżej  $280^\circ$ , podobnym do obserwowanego w rzeczywistości.

W badaniach sprawdzających wpływ presji poszukiwania pożywienia, agenty ewoluowały w kierunku dużej prędkości poruszania się oraz wąskiego pola widzenia o dalekim zasięgu. Ponadto, zauważono dwa rodzaje zachowań, które powstawały zależnie od sposobu dystrybucji żywności w środowisku. W przypadku losowego rozmieszczenia pojedynczych sztuk pokarmu, pierwszy agent zjada wszystko, co prowadzi do rywalizacji i braku korzyści wynikającej z grupowania się, czy podążania jeden za drugim. Pożywienie występujące rzadziej, ale w dużych skupiskach, wystarczających dla każdego, doprowadziło do wspólnego poruszania się agentów, co zwiększa obszar przeszukiwania i szansę znalezienia pokarmu.

W pracy [5], przeprowadzono badania nad ewolucją strategii ataków drapieżnika. W środowisku badawczym umieszczono ławicę ryb oraz jednego drapieżnika. Model sterowania agentami ofiar opiera się na algorytmie Craiga Reynoldsa z dodatkowym polem ucieczki. Wszystkie parametry algorytmu dla ryb są jednakowe, nie podlegają ewolucji i zostały dobrane w taki sposób, żeby wśród ofiar występowały zachowania stadne (formowanie ławicy). Agent drapieżnika posiada pełne pole widzenia ( $360^\circ$ ), a ataki obarczone są efektem dezorientacji. Jego zachowanie polega na losowym wyborze jednej z zaimplementowanych strategii:

1. Wybór jednej z prostych taktyk ataku, polegającej na obraniu za cel:
  - a) najbliższej ofiary,
  - b) ofiary najbliżej centrum stada,
  - c) ofiary najbardziej odizolowanej.
2. Rozproszenie ofiar, które odbywa się poprzez ściganie jednostki najbliższej centrum stada, przez określony dystans, a następnie wybraniu za cel najbardziej odizolowanej ofiary.

W przypadku drapieżnika, ewolucji podlegają wartości prawdopodobieństwa wyboru każdej z prostych strategii ataku, odległości po której przestaje rozpraszać ofiary, a także wielkości promienia poszukiwania najbardziej odizolowanej ofiary.

Przeprowadzone badania pokazały, że kiedy efekt dezorientacji drapieżnika nie jest obecny, najlepsze rezultaty osiąga się dzięki strategii atakowania najbliższej ofiary. W przeciwnym przypadku, strategia wcześniejszego rozproszenia stada jest bardziej skuteczna. Wyniki badań sugerują, że efekt dezorientacji drapieżnika może odgrywać ważną rolę w procesie ewolucji złożonych strategii polowania.

#### 1.4. BLIŻEJ RZECZYWISTOŚCI

Niewiele jest prac podejmujących próbę wiernego odtworzenia złożoności środowiska naturalnego. Zazwyczaj badania ograniczają się do pojedynczych aspektów życia zwierząt, pomijając takie elementy jak potrzeba zdobywania pożywienia, a sztuczne metody rozmnażania zastępują naturalny proces poszukiwania partnera w celach kopulacyjnych. Praca [22] jest jedną ze znalezionych, w której podjęto próbę odtworzenia wielu mechanizmów występujących w rzeczywistym środowisku. G. Werner oraz M. Dyer stworzyli "BioLand", czyli wirtualny świat zaprojektowany z myślą o eksperymentach nad ewolucją współpracy, współzawodnictwa i komunikacji. Głównym celem było stworzenie wystarczająco złożonej symulacji, która pozwoliłaby zaobserwować zachowania stadne wśród sztucznych zwierząt.

BioLand tworzy toroidalna przestrzeń, o wymiarze 1000 na 1000 wirtualnych metrów, zamieszkiwana przez dwa rodzaje zwierząt: jelenie (ofiary) oraz wilki (drapieżniki). Występują w nim również drzewa, będące przeszkodą w otoczeniu oraz rośliny, stanowiące pożywienie dla jeleni. Zwierzęta posiadają energię, którą mogą odnawiać spożywając odpowiedni dla siebie pokarm. Każda akcja, próba kopulacji i konsumpcji (ataku w przypadku wilków), poruszanie się, komunikacja, a nawet sama egzystencja, kosztuje organizm pewną porcję energii, a kiedy jej wartość spada do zera, osobnik ginie i jest usuwany z symulacji. Każdy obiekt, zarówno zwierzęta jak i rośliny, wydzielają specyficzny zapach, o intensywności malejącej wraz z odległością. Zwierzęta poruszając się wydają dźwięki o sile proporcjonalnej do swojej prędkości. Akcje związane z kopulacją, spożywaniem pokarmu oraz atakiem drapieżnika, również wydają specyficzne odgłosy. Oprócz tego, każdy osobnik może dobrowolnie wydać jeden z czterech dźwięków służących do komunikacji, różniących się częstotliwością i zasięgiem działania.

Zachowanie każdego zwierzęcia jest kontrolowane przez sztuczny mózg, będący prostą siecią neuronową. Agent posiada dwa receptory wykrywające bodźce wzrokowe, słuchowe oraz węchowe. Na podstawie wektora obserwacji, składającego się z prostych informacji odbieranych ze środowiska, takich jak "roślina po lewej/prawej", "samiec jelenia po lewej/prawej", "wilk po lewej/prawej", sieć podejmuje decyzje co do ruchu w danym kierunku, jedzenia/ataku, kopulacji lub wydania jednego z czterech rodzajów dźwięku. Dodatkowo, jeden neuron na wejściu sieci jest zawsze aktywny. Jego zadaniem jest napędzanie zwierząt do działania nawet w przypadku, kiedy nie odbierają one żadnych bodźców ze środowiska. Według autorów, takie zachowanie jest bardziej naturalne. Sieć posiada jedną warstwę ukrytą, składającą się z trzech neuronów, natomiast połączenia między nimi są dowolne i możliwe jest, aby neuron wejściowy miał bezpośrednie połączenie z warstwą wyjściową. Za ewolucję organizmów odpowiada naturalny proces selekcji oraz algorytm genetyczny. Nowy osobnik tworzony jest jedynie w przypadku, kiedy samiec i samica danego gatunku spotkają się, wykażą chęć kopulacji oraz będą mieli wystarczającą ilość energii. Potomstwo otrzymuje od rodziców krzyżówkę genów, z których pojedyncze mogą dodatkowo ulec mutacji.

Przeprowadzone eksperymenty pozwoliły zaobserwować proste zachowania stadne, spowodowane presją ataku ze strony drapieżnika, koniecznością szukania partnera do kopulacji oraz zwiększoną szansą zdobycia pożywienia. Gdy któryś z osobników wyczuje pokarm, zacznie poruszać się w jego kierunku, natomiast pozostałe jelenie w grupie podążą za nim. Ze strony drapieżników udało się zaobserwować wspólne polowania.

Warto zaznaczyć, że praca powstała w 1993 roku, w czasach, kiedy moc obliczeniowa była znacznie ograniczona. Z tego powodu, obecne symulatory wirtualnych środowisk mogłyby w większym stopniu oddać złożoność środowiska naturalnego, a zwierzęta być wyposażone w sterownik wykorzystujący sieć neuronową o bardziej rozbudowanej architekturze. Nie udało się zaleźć nowszych prac, które spróbowaliby chociaż w takim stopniu zbliżyć się do realizmu. Wygląda na to, że obecnie odchodzi się od tego typu eksperymentów, a prowadzone badania dotyczą pojedynczych, wybranych aspektów życia zwierząt.

## 1.5. PODEJŚCIE MATEMATYCZNE

Istnieją również prace, w których omawiany problem potraktowano w sposób bardziej formalny. Jedną z nich jest [25], autorstwa X. Wang i X. Zou, w którym wykorzystując metody analizy matematycznej oraz symulacji numerycznych, postanowiono zbadać w jaki sposób zachowania anty-drapieżnicze i związane z nimi koszty wpłyną na rozmieszczenie przestrzenne ofiar i drapieżników.

W tym celu wykorzystano szereg funkcji i zmiennych, m.in. gęstości stada zwierząt w danym położeniu i czasie oraz założono, że drapieżnicy poruszają się losowo. Ofiary natomiast, potrafią dostrzec ryzyko ataku drapieżnika i podjąć odpowiednie działania, aby unikać niebezpieczeństwa, co w konsekwencji prowadzi do ich ukierunkowanego ruchu na obszary z mniejszym prawdopodobieństwem ataku. Bazując na wcześniej przeprowadzonych badań biologicznych pokazujących, że sama świadomość ryzyka ataku zmniejszała liczbę potomstwa w ciągu roku o około 40%, do modelu wprowadzono koszt wynikający z obecności zachowań anty-drapieżniczych, odpowiadający zmniejszonej reprodukcji.

Udało się pokazać, że już niewielka wrażliwość ofiar na ryzyko ataku może doprowadzić do wykształcenia zachowań anty-drapieżniczych, a w konsekwencji do zwiększenia przeżywalności, podczas gdy zbyt duży koszt takich zachowań wykluczał ich powstanie.

## **1.6. PODSUMOWANIE**

Większość prac koncentruje swoje badania wokół presji wywieranej przez obecność drapieżnika. Brak natomiast prac uwzględniających w środowisku konieczność poszukiwania pożywienia, która jest niezwykle istotna w naturalnym środowisku i niekiedy może okazać się ważniejsza niż ochrona przed atakami. Jedyną znalezioną pracą, w której uwzględniono ten aspekt jest [24], lecz zastosowano tam dość ograniczony model sterowania agentami (z góry określony algorytm), a element zdobywania pożywienia badano niezależnie od presji drapieżnika.

Badania najczęściej prowadzono pod kątem ewolucji zachowań stadnych jako formy obrony przed drapieżnikami. Brak natomiast eksperymentów, które pozwoliłyby na swobodną ewolucję drapieżników i obserwacje, czy uda im się wykształcić strategie polowania przełamujące defensywę ofiar. Podobne symulacje były przeprowadzane w [5], jednak drapieżnik nie mógł dowolnie dostosować swoich zachowań, ponieważ był ograniczony do wyboru jednej z już zaimplementowanych strategii.

Pomimo dużych możliwości obliczeniowych, oferowanych przez współczesną technologię, zazwyczaj jako sterownik sztucznych organizmów wykorzystywano dość proste modele, bazujące na sieci Markowa, czy klasycznych rekurencyjnych sieciach neuronowych. Obecne możliwości głębokich sieci neuronowych są znacznie większe, dlatego wykorzystanie ich jako sterownika, mogłoby przyczynić się do wykształcenia ciekawszych i bardziej złożonych zachowań zwierząt. Nie powstają również symulacje, próbujące jak najwierniej odtworzyć warunki panujące w naturalnym środowisku. Znalezione prace ograniczały się jedynie do dwóch gatunków (ofiar i drapieżników).

Powyższe braki stanowiły motywację do napisania tej pracy i przeprowadzenia badań wykorzystując i porównując bardziej złożone sieci neuronowe, a także opracowania rozbudowanego środowiska i uwzględnienia w badaniach zarówno presji drapieżnika jak i konieczności zdobywania pożywienia. Oprócz tego, zostanie wprowadzony mechanizm komunikacji, który względem pracy [12], rozszerzono do dwóch rodzajów dźwięków, a także obecny będzie zarówno u agentów ofiar, jak i drapieżników, dzięki czemu, być może uda się zaobserwować interesujące zachowania współpracy oraz strategie ataków.

## 2. WIRTUALNY ŚWIAT

Na potrzeby badań przeprowadzonych w niniejszej pracy, wymagane było stworzenie odpowiedniej symulacji. Niniejszy rozdział zawiera teoretyczny opis środowiska wykorzystanego w tym celu, wcześniej jednak próbuje przybliżyć pojęcie sztucznego życia oraz ideę tworzenia wirtualnych światów.

### 2.1. KONCEPCJA SZTUCZNEGO ŻYCIA

Ludzie od wieków zastanawiali się, czym tak właściwie jest życie. Co jest niezbędne, aby mogło istnieć i rozmnażać się? Co doprowadziło do rozwinięcia się tak złożonych organizmów? Jakie mechanizmy odpowiadają za ich inteligentne zachowania oraz umożliwiają adaptację do zmian zachodzących w środowisku?

Życie na Ziemi jest niezwykle złożone. Biologiczne organizmy zbudowane są z niewyobrażalnej liczby, szerokiej gamy różnych związków chemicznych oraz żywych komórek, które cyklicznie umierają i odradzają się, tworząc własny, wewnętrzny i niezwykle skomplikowany system. Miliony gatunków, posiadających najrozmaitsze kształty i rozmiary oraz przejawiających często niebywałe zachowania, oddziałują na siebie wzajemnie w środowisku naturalnym na niezliczoną ilość sposobów. W tym niezbadanym, lecz tylko pozornym chaosie, trudno jest wyodrębnić pewne cechy fundamentalne dla życia jako ogólnego zjawiska, od tych będących jedynie dziełem przypadku w procesie ewolucji. Zegara ewolucyjnego nie sposób cofnąć, żeby zobaczyć, które cechy pojawiają się zawsze i są nieodłącznym aspektem życia. Nie istnieją również żadne obce (pozaziemskie) ekosystemy, które można by porównać. W ostatnim czasie, z pomocą przychodzą wirtualne środowiska, których wykorzystanie w celu badania procesów powstawania oraz ewolucji sztucznego życia, począwszy od najprostszych jego form, może okazać się pomocne w zrozumieniu samej koncepcji życia, a także odpowiedzi na wyżej postawione pytania.

Najpierw jednak, należy się zastanowić nad definicją sztucznego życia. Uznaje się, że pierwszy jego model powstał w 1951 roku, kiedy John von Neumann starał się zrozumieć podstawowe właściwości życia, w szczególności interesując się samoreplikacją. Zdefiniował pojęcie automatu komórkowego oraz zaproponował samoreplikujący się system formalny, którego celem miała być uniwersalność obliczeniowa (in. *Kompletność Turinga*) oraz zdolność do swobodnej, nieograniczonej ewolucji.

Termin sztucznego życia (ang. *Artifical Life, ALife*) [7], znany obecnie, został sformułowany pod koniec lat osiemdziesiątych przez Langtona, który zdefiniował je jako "życie stworzone przez człowieka, a nie przez naturę", czyli jest to nauka o systemach stworzonych przez człowieka, wykazujących zachowania charakterystyczne dla naturalnych systemów życia. Według innej definicji, sztuczne życie można podzielić na trzy przeplatające się kategorie, odpowiadające trzem różnym metodom jego syntezy. "Miękkie" sztuczne życie, tworzą symulacje lub inne czysto cyfrowe konstrukcje, które wykazują zachowania charakterystyczne dla życia, "twarde" tworzą sprzętowe wdrożenia systemów, natomiast "mokre" otrzymuje się z substancji biochemicznych. W tym znaczeniu, tworząc systemy posiadające właściwości zbliżone do życiowych, a także rozwijając nowe jego formy, dążą się do lepszego zrozumienia biologicznego życia.

Obecnie, ALife stało się interdyscyplinarną dziedziną badań, które według pracy [2], można podzielić na 14 różnych obszarów. Każdy z nich zajmuje się nieco innym aspektem sztucznego życia, inaczej je również definiując. Granice między nimi nie są jednak sztywne i zdarza się, że poszczególne dziedziny przeplatają się ze sobą. W odniesieniu do niniejszej pracy, sztuczne życie w niej przedstawione, najlepiej może zostać w opisane kategorie samoorganizacji oraz adaptacji.

W określeniu samoorganizacji, sztuczne życie jest zjawiskiem, w którym lokalne interakcje między pojedynczymi elementami, prowadzą do powstania globalnych wzorców lub zachowań, między innymi takich jak w rojach czy stadach. Wczesne przykłady samoorganizacji w kontekście sztucznego życia to na przykład algorytm "Boidów" [17], który jest przykładem modelu tworzenia ruchu zbiorowego, wynikającego z prostych reguł jednostkowych.

W kontekście adaptacji, która jest jednym z najważniejszych obszarów sztucznego życia, jest ono rozumiane jako proces zmian agenta lub systemu, które w odpowiedzi na stan otaczającego środowiska, pomogą mu w osiągnięciu jego własnych celów. Adaptacja jest główną cechą żywych systemów, niezbędną dla ich autonomii oraz przetrwania. Może ona zachodzić w różnych skalach czasowych. W powolnej skali, obejmującej kilka okresów życia (pokoleń), jest znana jako proces ewolucji. Stanowił on inspirację do opracowania algorytmów genetycznych, które później uogólnione zostały w dziedzinie obliczeń ewolucyjnych. Głównym celem ich wykorzystania jest poszukiwanie właściwych rozwiązań w przestrzeniach złożonych, które są trudne do zbadania bardziej tradycyjnymi metodami heurystycznymi. Skala średnia, obejmująca pojedyncze życie, traktuje adaptację jako rozwój fizyczny oraz poznawczy. W szybkiej skali, stanowiącej ułamek całego życia, nazywana jest procesem uczenia się, rozumianym jako zmiana zdolności lub zachowania organizmu spowodowana przez doświadczenie. Jest on podstawowym aspektem zachowań adaptacyjnych organizmów żywych. W kontekście sztucznego życia przyjęto kilka podejść do modelowania uczenia się, z których najczęściej stosowane są sztuczne sieci neuronowe, inspirowane strukturą i działaniem biologicznych sieci komórek nerwowych w mózgu.

Badania zjawisk zachodzących w przyrodzie, studiowanie życia czy procesów ewolucyjnych, stanowiły inspirację oraz przyczyniły się do stworzenia wielu metod i narzędzi, powszechnie wykorzystywanych w różnych dziedzinach nauki. Przyszedł jednak czas, w którym role się odwracają i te same metody (szczególnie algorytmy ewolucyjne oraz sieci neuronowe), są wykorzystywane do symulacji procesów życiowych. Niesamowity postęp technologiczny ostatnich lat, pozwala na tworzenie coraz bardziej złożonych, lecz wciąż wysoce uproszczonych, sztucznych środowisk, które służą do badania dynamicznych zachowań obiektów lub systemów w odpowiedzi na warunki, które nie mogą być w bezpieczny czy prosty sposób wprowadzone w rzeczywistości. Sztuczne życie jest przedmiotem wielu badań w informatyce, a ich wyniki są z powodzeniem stosowane również w wielu innych dziedzinach nauki [11]. Studiowanie sztucznego życia oraz procesu jego powstawania, może ponadto przyczynić się do odkrycia czegoś o kluczowym znaczeniu, dzięki czemu możliwe będzie lepsze zrozumienie genezy oraz istoty biologicznego życia, jakie obserwuje się na co dzień.

## 2.2. ŚRODOWISKO BADAWCZE

Wirtualne światy powstają z zamysłem odtworzenia pewnych aspektów rzeczywistości, w zakresie wymaganym dla danego problemu. Nic jednak nie stoi na przeszkodzie przed zbudowaniem zupełnie abstrakcyjnych światów o niespotkanych dotąd właściwościach. Zazwyczaj znajdują one zastosowanie w filmach i grach (które same w sobie są symulacją), ale coraz częściej są również wykorzystywane do prowadzenia badań naukowych. W tak kontrolowanych warunkach, jakie zapewnia środowisko symulacji, można stosunkowo łatwo zmieniać jego parametry i reguły, a także odizolować pewne wybrane czynniki oraz obserwować ich wpływ na pozostałe elementy.

W dziedzinie badań nad zachowaniem zwierząt, a w szczególności nad tworzeniem się zachowań stadnych, co jest tematem tej pracy, sprowadza się to do zaprojektowania otwartego środowiska, zazwyczaj w postaci dwuwymiarowej planszy oraz osobników (agentów), które zostaną w nim umieszczone. Takie agenty najczęściej są sterowane za pomocą specjalnego modelu, pełniącego rolę sztucznego mózgu, który umożliwia im względnie samodzielne podejmowanie decyzji (np. o poruszaniu się, czy wykonaniu pewnych akcji), na podstawie odbieranych ze środowiska bodźców. Istnieją jednak przypadki, w których ta swoboda jest im odebrana, na rzecz z góry narzuconego algorytmu, określającego ich zachowanie [5, 24]. Agenty są wyposażone w zmysły (np. wzroku, słuchu, węchu), dzięki którym mogą reagować na zmiany zachodzące w środowisku, a dodatkowo mogą posiadać zdolności takie jak mechanizm komunikacji, czy spożywanie pokarmu. Niekiedy nakładane są również pewne ograniczenia w postaci efektu dezorientacji, energii życiowej organizmów, czy upływu czasu (procesu starzenia się). Zależnie od potrzeb, w środowisku mogą zostać uwzględnione dodatkowe elementy, takie jak rośliny (np. jako forma pożywienia), czy różnego rodzaju przeszkody (np. skały, rzeki). Głównym zadaniem agentów jest przetrwanie i dostosowanie się do warunków panujących w środowisku, które ma związek z unikaniem niebezpieczeństwa (np. drapieżników), poszukiwaniem partnerów do kopulacji, czy zdobywaniem pożywienia. Agenty uczą się niezbędnych zachowań dzięki procesowi sztucznej ewolucji, symulowanej za pomocą algorytmu genetycznego.

Środowisko zaprojektowane na potrzeby niniejszej pracy, jest dwuwymiarową, toroidalną przestrzenią, zamieszkiwaną przez dwa gatunki stworzeń (agentów), należące do grupy ofiar oraz drapieżników. Podczas trwania symulacji, w środowisku może występować wiele przedstawicieli obydwu z nich. Kontrolerem każdego osobnika jest model rekurencyjnej sieci neuronowej. W celach porównawczych, zostały wykorzystane dwa takie modele, RNN oraz LSTM.

Agenty wyposażone zostały w zmysł wzroku oraz słuchu. Posiadają również zdolność konsumpcji pokarmu, a także mechanizm komunikacji. Ofiary mają możliwość wydawania dwóch rodzajów dźwięków, ostrzegawczego (przy pojawienniu się drapieżnika) oraz informacyjnego (o znalezieniu pożywienia). Drapieżnicy posiadają tylko jeden dźwięk, nie ma on natomiast przypisanego żadnego konkretnego znaczenia, dlatego mogą nadać je sami i nauczyć się z niego korzystać wedle własnych potrzeb.

Organizm ofiar posiada energię życiową, którą traci wraz z upływem czasu. Kiedy wartość energii spadnie do zera, dany osobnik ginie i jest usuwany z symulacji. W środowisku występują rośliny, które stanowią pożywienie dla ofiar i pozwalają odnawiać zasoby energii. Zarówno kąt jak i zasięg zmysłu wzroku i słuchu, są identyczne u obu gatunków. Drapieżnicy mają jednak przewagę nad ofiarami pod względem szybkości poruszania się, natomiast te, są z kolei bardziej zwinne (większa szybkość skrętu). Każdy drapieżnik, dodatkowo jest obarczony efektem dezorientacji, która zmniejsza skuteczność ataków podczas polowania na grupy ofiar.

Głównym zadaniem agentów jest wykształcenie zachowań, które pomogą przetrwać w środowisku. Muszą się one skupić na poszukiwaniu pożywienia, a ofiary dodatkowo na unikaniu niebezpieczeństwa ze strony drapieżników. Proces ewolucji, odpowiedzialny za uczenie się agentów, został zamodelowany z wykorzystaniem algorytmu genetycznego.

W celu sensownego i wiarygodnego porównania wyników otrzymanych w tej pracy, z tymi uzyskanymi w artykułach [1, 12, 15, 16], zastosowano takie same wartości dla najważniejszych parametrów środowiska. Informacje dotyczące szczegółów implementacyjnych wirtualnego środowiska oraz jego elementów składowych, zostały opisane w rozdziale 4.

### **3. ZASTOSOWANE MODELE**

Sztuczne życie zamieszkujące wirtualne środowisko, musi posiadać względną autonomię oraz potrafić samodzielnie podejmować decyzje, które pozwolą na przetrwanie w zmieniających się warunkach. W rzeczywistości, zwierzęta uczą się tych zachowań dzięki zdobytemu doświadczeniu oraz cechom przekazywanym w genach kolejnym pokoleniom. Agenty muszą być zatem wyposażone w pewnego rodzaju "sztuczny mózg", który dzięki symulacji procesu ewolucji, pozwoli na wykształcenie cech i zachowań niezbędnych do egzystencji w środowisku.

Niniejszy rozdział zawiera teoretyczny opis modeli sieci neuronowych, wykorzystanych jako sterownik agentów oraz algorytmu genetycznego imitującego proces naturalnej ewolucji.

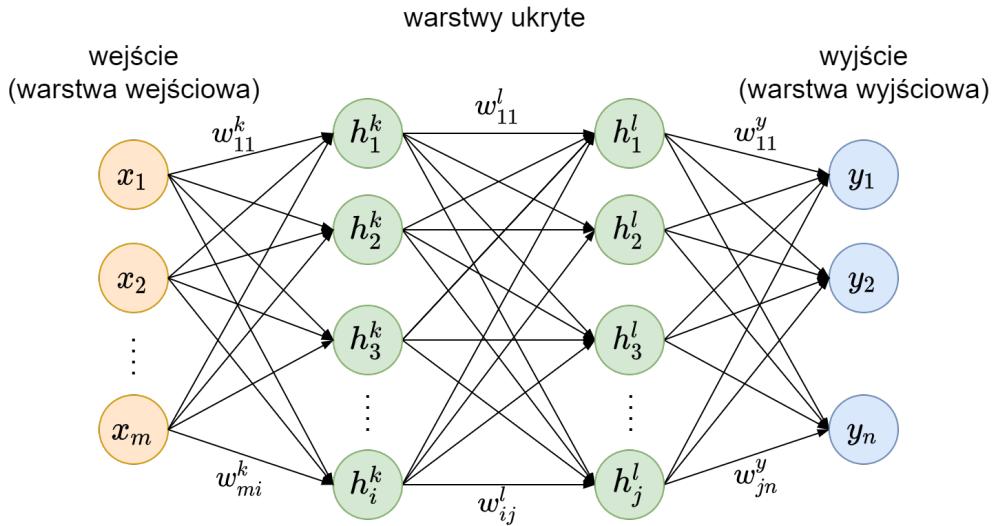
#### **3.1. SZTUCZNA SIEĆ NEURONOWA**

Komputery przewyższają ludzkie zdolności w wielu kwestiach. Potrafią w ułamku sekund przeprowadzić obliczenia na wielkich liczbach, zachowując dużą dokładność czy przeszukać ogromne zasoby danych. Ludzki mózg wciąż jednak pozostaje niedościgniony w kategoriach zdrowego rozsądku, inspiracji i wyobraźni. Potrafi również interpretować kontekst rzeczywistych sytuacji w sposób nieosiągalny przez współczesną technologię. Inspirowane jego strukturą i działaniem, sztuczne sieci neuronowe [6, 13] (ang. *Artifical Neural Networks, ANN*), są próbą stworzenia systemu, który będzie w stanie uczyć się i podejmować decyzje w sposób przypominający ludzki.

Żywy mózg zbudowany jest z miliardów komórek nerwowych (neuronów) oraz nieporównywalnie większej liczby połączeń między nimi, tworząc najbardziej skomplikowaną strukturę znaną człowiekowi. Neurony odbierają sygnały pochodzące z innych komórek i jeśli wypadkowa ich sił będzie wystarczająca, impuls może zostać przekazany dalej. Połączenia między neuronami mogą stać się silniejsze lub słabsze, mogą także powstać nowe połączenia, podczas gdy inne mogą zaniknąć. Ten proces jest odpowiedzialny za uczenie się oraz pamięć. Każda prosta informacja czy myśl, zanim stanie się rzeczywistością, musi pokonać długą drogę i przejść przez tysiące komórek nerwowych.

Sposób działania sztucznej sieci neuronowej jest podobny, lecz znacznie uproszczony względem naturalnych procesów zachodzących w mózgu. W celu przetworzenia oraz nadaniu znaczenia informacji wejściowej, wykorzystuje aparat matematyczny. Pojedynczy neuron posiada zestaw połączeń z przypisanymi wagami, które zależnie od wartości tłumią lub wzmacniają poszczególne sygnały. Wypadkowa wartość aktywacji neuronu jest liczona jako suma ważona siły sygnałów wejściowych oraz wag im odpowiadających. Do tej wartości zazwyczaj dodawany jest jeszcze bias, a całość przetwarza jest przez funkcję aktywacji. W sposób formalny przedstawia to wzór (3.1), gdzie  $w_j$  jest wagą  $j$ -tego połączenia,  $x_j$  jest  $j$ -tą wartością wejściową,  $b$  wartością biasu, natomiast  $f_a$  jest funkcją aktywacji.

$$\begin{aligned} z &= \sum_j w_j x_j + b \\ a &= f_a(z) \end{aligned} \tag{3.1}$$



Gdzie:

- $x_m$  -  $m$ -ta wartość warstwy wejściowej
- $y_n$  -  $n$ -ta wartość warstwy wyjściowej
- $h_i^k$  -  $i$ -ty neuron warstwy  $k$
- $h_j^l$  -  $j$ -ty neuron warstwy  $l$
- $w_{ij}^l$  - waga połączenia między  $j$ -tym neuronem warstwy  $l$ , a  $i$ -tym neuronem warstwy  $k$

Rys. 3.1. Schemat sztucznej sieci neuronowej

Przeważnie ANN zbudowana jest z setek lub nawet milionów sztucznych neuronów, ułożonych w szeregu warstw (Rys. 3.1), w taki sposób, że każdy neuron danej warstwy jest połączony ze wszystkimi neuronami warstw sąsiadujących. Uogólnienie wzoru 3.1 dla całej warstwy neuronów zostało przedstawione na wzorze 3.2. Dla uproszczenia stosuje się zapis macierzowy.

$W_i$  oraz  $b_i$  są wektorami zawierającymi odpowiednio wagę oraz biasy każdego neuronu z  $i$ -tej warstwy,  $x$  jest wektorem zawierającym wartości wejściowe do sieci lub wyjście poprzedniej warstwy,  $f_a$  jest funkcją aktywacji,  $z_i$  jest wektorem pobudzenia każdego neuronu  $i$ -tej warstwy, natomiast  $a_i$  jest wektorem aktywacji przetworzoną przez funkcję  $f_a$ .

$$\begin{aligned} z_i &= W_i \cdot x + b_i \\ a_i &= f_a(z_i) \end{aligned} \tag{3.2}$$

Sieć neuronowa tworząca biologiczny mózg, uczy się poprzez zdobywane doświadczenie i powtarzanie pewnych zachowań, które powodują powstawanie zupełnie nowych połączeń neuronowych oraz pogrubianie tych już istniejących. W sztucznych sieciach neuronowych zachodzi podobny proces, polegający na odpowiednim dostosowywaniu wartości wag połączeń między poszczególnymi neuronami. Istnieje wiele metod nauki ANN, jedną z najczęściej stosowanych jest metoda propagacji wstecznej, w której na podstawie otrzymanego wyjścia sieci oraz wartości oczekiwanych, wyliczany jest błąd służący do manipulacji wartościami wag. Sieć uczy się, zmniejszając tym samym błąd na wyjściu.

W niektórych przypadkach, kiedy wartości wyjścia z sieci nie mogą być jednoznacznie określone, takie podejście nie może zostać zastosowane. Taka sytuacja ma miejsce w problemie sztucznego życia, podjętego w tej pracy. Agenty sterowane siecią neuronową, na podstawie bodźców odbieranych ze środowiska (wektora obserwacji), podejmują decyzję co do następnego ruchu. Nie można jednak jednoznacznie określić, jedynego poprawnego zachowania w danej sytuacji. Decyzje podejmowane przez agentów mogą być rozpatrywane jedynie w perspektywie czasowej oraz skutków jakie przyniosły. Stosowane są wówczas inne podejścia, na przykład wykorzystujące algorytm genetyczny, który wymaga jedynie ogólnej oceny sprawności danego osobnika (agenta), uzyskiwanej podczas przeprowadzania symulacji. Opis algorytmu genetycznego znajduje się w podrozdziale 3.4.

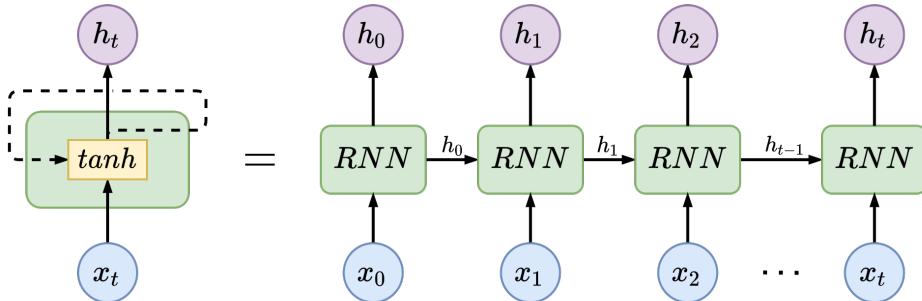
### 3.2. REKURENCYJNA SIEĆ NEURONOWA

Struktura sieci neuronowej, dzięki odpowiednio dobranym wagom, odpowiada za podstawowe zachowania agentów. Zmudną drogą losowych zmian (często niekorzystnych), powstają cechy, które przekazywane kolejnym pokoleniom drogą selekcji naturalnej, prowadzą do powstania zachowań umożliwiających przetrwanie w wirtualnym środowisku. To zjawisko nazywane jest instynktem, czyli wrodzoną (odziedziczoną wraz z genami) zdolnością wykonywania podstawowych czynności istotnych dla przetrwania. W naturze, zwierzęta nie kierują się wyłącznie instynktem, lecz także wiedzą oraz doświadczeniem zgromadzonym przez całe życie. Ponadto, nie podejmują decyzji wyłącznie pod wpływem danej chwili. Ich umysł działa na podstawie bodźców odbieranych za pośrednictwem zmysłów, ale również wykorzystując kontekst sytuacji oraz przechowywane w pamięci informacje z przeszłości.

Klasyczne sieci neuronowe nie oferują takich możliwości. Zakładają, że wszystkie informacje wejściowe (i odpowiadające im wyjścia) są od siebie niezależne. Stanowi to poważne ograniczenie i powoduje, że sieć nie sprawdza się zbyt dobrze w zadaniach, w których dane występują w formie sekwencyjnej. Naśladowując ludzki umysł, zdolny do wykorzystania wcześniej zdobytej informacji w odpowiednim czasie, opracowano rekurencyjną sieć neuronową [6, 18] (ang. *Recurrent Neural Network, RNN*). W przeciwieństwie do poprzednio opisanych, klasycznych sieci, w których przepływ informacji odbywa się w sposób jednokierunkowy, w sieciach rekurencyjnych występują sprzężenia zwrotne, które umożliwiają przekazywanie informacji między kolejnymi cyklami przetwarzania. W każdym kroku czasowym, wyjście sieci wyliczane jest na podstawie bieżących danych wejściowych oraz dotychczasowych obliczeń wykonanych na poprzednich elementach z sekwencji. Umożliwia to przechowanie przetworzonych informacji oraz wykorzystywanie ich w odpowiedniej chwili w przeszłości. Schemat działania RNN został przedstawiony na rysunku 3.2. Sieci rekurencyjne znalazły zastosowanie w przetwarzaniu języka naturalnego, wideo oraz innych zadaniach wymagających uwzględnienia aspektu czasowego, dlatego dobrze sprawdzą się również w roli sterownika sztucznych zwierząt.

Działanie RNN można opisać jednym prostym wzorem (3.3), gdzie  $h_t$  jest wyjściem z sieci w bieżącym kroku czasowym,  $W_{hh}$  jest macierzą wag stanu wewnętrznego,  $W_{xh}$  macierzą wag stanu wejściowego,  $b_h$  wektorem biasów sieci,  $x_t$  jest aktualnym stanem wejściowym sieci, a  $\tanh$  funkcją tangens hiperboliczny. W pierwszym cyklu przetwarzania, dla  $t=0$ , wartości wektora  $h_{t-1}$  są równe 0.

$$h_t = \tanh(W_{hh} \cdot h_{t-1} + W_{xh} \cdot x_t + b_h) \quad (3.3)$$



Gdzie:

- $x_t$  - wektor danych wejściowych w kroku  $t$
- $h_t$  - wektor wyjściowy sieci w kroku  $t$
- $\dashrightarrow$  - linia obrazująca pętlę czasową sieci
- $\text{tanh}$  - warstwa neuronów z funkcją aktywacji tangens hiperboliczny

Rys. 3.2. Pętla sieci RNN oraz jej rozwinięcie w czasie

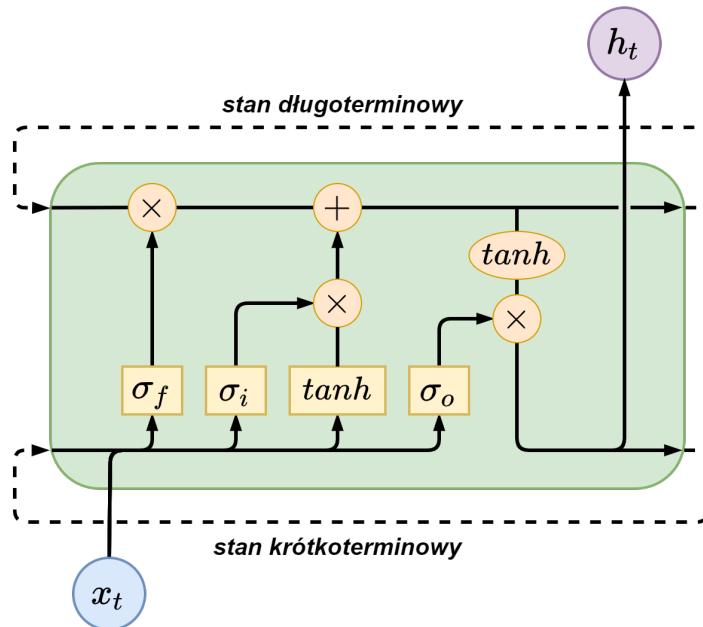
Sterownik zaprojektowany na potrzeby niniejszej pracy, na podstawie wektora obserwacji musi zwrócić akcje agenta, dlatego oprócz sieci rekurencyjnej posiada dodatkową warstwę neuronów, która w danym kroku czasowym przekształca wyjście  $h_t$  sieci RNN na wektor akcji  $a_t$ . Zostało to przedstawione na wzorze (3.4), gdzie  $W_a$  jest macierzą wag tej warstwy, natomiast  $b_a$  wektorem wag jej biasów. Więcej informacji na temat budowy sterownika znajduje się w podrozdziale 4.1.

$$a_t = \tanh(W_a \cdot h_t + b_a) \quad (3.4)$$

### 3.3. SIEĆ REKURENCYJNA LSTM

Jednym z kluczowych aspektów sieci rekurencyjnych jest ich potencjał do wykorzystania informacji uzyskanych w poprzednich krokach, na późniejszym etapie. Niekiedy wystarczy odnieść się do najświeższych informacji, ale są też przypadki, w których potrzebny jest znacznie szerszy kontekst. Może się zdarzyć, że odległość czasowa między informacją, a punktem w którym jest ona potrzebna, stanie się bardzo duża. Teoretycznie, RNN powinna poradzić sobie z takimi długoterminowymi zależnościami, jednak w praktyce, w miarę jak luka czasowa rośnie, nie jest w stanie nauczyć się jak powiązać tak odległe informacje, a jej możliwości często ograniczają się do zaledwie kilku kroków czasowych wstecz.

Ludzki umysł pozwala na zapamiętanie wielu informacji i wykorzystywanie ich po długim czasie. Wyposażony jest w pamięć krótkotrwałą, która na pewien krótki okres czasu, zapisuje szczegółowe dane pochodzące z ostatnich chwil, pozwalając uchwycić kontekst czasowy danej sytuacji. W pamięci długotrwałej z kolei, informacje mogą być przechowywane nawet całe życie. Nie można jednak zapamiętać ich zbyt wiele, co dotyczy również sieci rekurencyjnej. Zamiast gromadzić całą przeszłość, lepszym rozwiązaniem jest wybiórcze zapamiętywanie wyłącznie istotnych informacji, zapominając o pozostałych. Idąc tym kierunkiem, opracowano sieci długoterminowej pamięci krótkotrwałej [14, 18] (ang. *Long Short-Term Memory, LSTM*), które są szczególnym rodzajem sieci rekurencyjnych, zaprojektowanym specjalnie w celu uniknięcia problemu długoterminowych zależności. Zapamiętywanie i przechowywanie informacji przez długi okres czasu jest praktycznie ich domyślnym zachowaniem, dlatego są obecnie szeroko stosowane w wielu różnych problemach.



Gdzie:

- $x_t$  - wektor danych wejściowych w kroku  $t$
- $h_t$  - wektor wyjściowy sieci w kroku  $t$
- $(+ \times)$  - suma wektorów oraz iloczyn hadamarda
- $\tanh$  - funkcja tangens hiperboliczny
- $\sigma_f$  - warstwa neuronów bramki zapominania z sigmoidalną funkcją aktywacji
- $\sigma_i$  - warstwa neuronów bramki wejściowej z sigmoidalną funkcją aktywacji
- $tanh$  - warstwa neuronów bramki wejściowej z funkcją aktywacji tangens hiperboliczny
- $\sigma_o$  - warstwa neuronów bramki wyjściowej z sigmoidalną funkcją aktywacji

Rys. 3.3. Schemat wewnętrznej budowy sieci LSTM

Wszystkie sieci rekurencyjne przybierają formę łańcucha następujących po sobie iteracji, a różnica między nimi tkwi w ich strukturze wewnętrznej. Sposób działania RNN można zawsze zredukować do jednego, prostego wzorca (3.3). W porównaniu do niej, budowa sieci LSTM przedstawiona na rysunku 3.3, jest znacznie bardziej złożona. Charakteryzuje się dwoma wewnętrznymi "stanami", które odpowiadają za zależności w kolejnych krokach czasowych. Stan krótkoterminowy działa podobnie jak w przypadku RNN i pozwala na przechowywanie bieżących informacji. Stan długoterminowy przypomina pamięć zewnętrzną, do której zapisywane są wyłącznie wybrane informacje. LSTM wykorzystuje specjalne struktury nazywane bramkami (ang. gates), które biorąc pod uwagę bieżące dane wejściowe oraz stan krótkoterminowy z poprzedniego kroku czasowego, modyfikują przechowywane w nim informacje. Istnieją trzy tego typu bramki, każda odpowiedzialna za inne zadanie. Bramka "zapominania" (ang. forget gate) decyduje o których danych zapomnieć i w jakim stopniu. Bramka "wejściowa" (ang. input gate) odpowiada za nowe informacje, które warto zapisać. Bramka "wyjściowa" (ang. output gate) podejmuje decyzję co do wartości, jakie powinny znaleźć się na wyjściu sieci.

Działanie modelu LSTM opisują równania (3.5).  $f_t$  jest wyjściem bramki zapominania w kroku  $t$ , natomiast  $W_{hf}, W_{xf}, b_f$  są wagami jej neuronów. Analogicznie,  $o_t$  jest wyjściem bramki wyjściowej, a  $W_{ho}, W_{xo}, b_o$  są jej wagami. Bramka wejściowa posiada dwa wyjścia  $i_t$  i  $C'_t$  oraz odpowiadające im wagie  $W_{hi}, W_{xi}, b_i$  i  $W_{hC}, W_{xC}, b_C$ .  $\sigma$  jest funkcją sigmoidalną, a  $\tanh$  funkcją tangens hiperboliczny.  $C_t$  jest nowym stanem długoterminowym w kroku  $t$ , a  $h_t$  jest wyjściem z sieci oraz stanem krótkoterminowym w kroku  $t + 1$ .

$$\begin{aligned}
 f_t &= \sigma(W_{hf} \cdot h_{t-1} + W_{xf} \cdot x_t + b_f) \\
 i_t &= \sigma(W_{hi} \cdot h_{t-1} + W_{xi} \cdot x_t + b_i) \\
 C'_t &= \tanh(W_{hC} \cdot h_{t-1} + W_{xC} \cdot x_t + b_C) \\
 o_t &= \sigma(W_{ho} \cdot h_{t-1} + W_{xo} \cdot x_t + b_o) \\
 C_t &= f_t * C_{t-1} + i_t * C'_t \\
 h_t &= o_t * \tanh(C_t)
 \end{aligned} \tag{3.5}$$

Analogicznie jak w przypadku RNN, wyjście z sieci LSTM  $h_t$  jest przekształcane na wektor akcji  $a_t$ , według wzoru 3.4.

### 3.4. ALGORYTM GENETYCZNY

Darwinizm, czyli ogólna teoria rozwoju życia, sformułowana przez brytyjskiego przyrodnika Karola Darwina, stała się fundamentem nauki o ewolucji. Stwierdza, że wszelkie gatunki organizmów powstają i rozwijają się, dzięki selekcji naturalnej oraz dziedzicznym cechom, które zwiększą zdolność jednostki do rywalizacji, przetrwania i rozmnażania się. Ewolucja tą drogą stanowi główną ideę w biologii, a poznanie jej koncepcji w ogromnym stopniu wpłynęło na pogląd dotyczący systemów biologicznych. Badania poświęcone tym zagadnieniom stanowiły inspirację do stworzenia algorytmów ewolucyjnych [19] (ang. *Evolutionary algorithm, EA*).

Z dostępnych obecnie metod ewolucyjnych, najszerzej stosowany, a równocześnie najprostszy jest algorytm genetyczny [8] (ang. *Genetic algorithm, GA*), z powodzeniem wykorzystywany przede wszystkim w problemach optymalizacyjnych. Jego działanie oparte jest na idei przetrwania najbardziej przystosowanych osobników, która wprost zaczerpnięta została z teorii ewolucji Darwina. W związku z tym, doskonale nadaje się również do modelowania procesów ewolucyjnych w sztucznych systemach życia, które po częścią są także problemem optymalizacyjnym, polegającym na znalezieniu osobników najbardziej przystosowanych do egzystencji w określonym środowisku. Algorytm genetyczny, zupełnie jak naturalny proces ewolucji, w głównej mierze podlega czynnikowi przypadku, w którym losowe, niewielkie zmiany wprowadzane do obecnej puli rozwiązań, stopniowo prowadzą do uzyskania coraz lepszych rezultatów.

Schemat działania GA można przedstawić w następującym cyklu:

1. Rozpoczęcie od losowo wygenerowanej populacji osobników (kandydatów do rozwiązania problemu).
2. Obliczenie wartość funkcji przystosowania każdego osobnika w populacji.
3. Zastosowanie metod selekcji oraz operatorów genetycznych (krzyżowania i mutacji) do bieżącej populacji w celu utworzenia nowego pokolenia.
4. Jeśli w populacji znajduje się rozwiązanie spełniające zadane kryterium lub jeśli osiągnięto maksymalną liczbę iteracji algorytmu (generacji), GA kończy działanie. W przeciwnym przypadku następuje powrót do kroku 2.

Działanie algorytmu genetycznego rozpoczyna się od wygenerowania początkowej populacji, składającej się z określonej liczby osobników. Każdy z nich posiada genotyp, czyli charakterystyczny zbiór wartości (genów), który reprezentuje pewną propozycję rozwiązania dla poruszанego problemu (niekoniecznie dobrą). Istnieją różne formy reprezentacji genotypu. Wybór właściwej zależy od specyfiki problemu i może zmniejszyć wymiar przestrzeni parametrów wymaganych do przeszukania, a tym samym ułatwić i przyspieszyć znalezienie zadowalającego rozwiązania. Genotyp może mieć postać binarną, w której każdy gen przyjmuje wartość równą 0 lub 1, permutacyjną, gdzie zmianie ulega kolejność genów (wartości pozostają stałe), czy kodować rzeczywiste wartości parametrów w sposób bezpośredni.

Każdy osobnik posiada wartość przystosowania (wartość fitness), czyli ocenę jakości rozwiązania, które reprezentuje. Jest ona określana z wykorzystaniem specjalnej funkcji modelującej rozważany problem lub jak w przypadku dziedziny sztucznego życia, dzięki przeprowadzeniu symulacji pozwalającej na sprawdzenie jak dany osobnik (agent) radzi sobie w wirtualnym środowisku. Wykorzystując wartość fitness, algorytm genetyczny dokonuje selekcji jednostek, które zostaną rodzicami i przekażą swoje geny następnemu pokoleniu. Jednostki lepiej przystosowane mają na to większą szansę, jednak zależnie od podejścia, dopuszcza się również wybór słabszych osobników w celu zachowania większej różnorodności genetycznej. W niniejszej pracy wykorzystano kilka różnych metod selekcji, przedstawionych poniżej. Przeprowadzając odpowiednie badania (rozdział 5) wybrano podejście, przy którym proces uczenia się zachodzi najskuteczniej.

Metody selekcji osobników:

- Metoda koła ruletki (ang. *Roulette Wheel selection*)

Selekcja  $n$  jednostek następuje losowo, przy czym prawdopodobieństwo wyboru każdego z osobników w populacji jest stosunkiem wartości jego funkcji przystosowania do sumy tych wartości całej populacji.

- Metoda turniejowa (ang. *Tournament selection*)

Z populacji, losowo wybranych zostaje  $k$  osobników, którzy biorą udział w turnieju. Spośród nich wygrywa najbardziej sprawny kandydat. W ten sposób odbywa się  $n$  takich turniejów, z których pochodzi ostatecznie wyselekcjonowana grupa jednostek.

- Metoda rankingowa (ang. *Rank selection*)

Osobniki w populacji sortowane są według wartości ich oceny, a następnie wybieranych jest  $n$  najlepszych jednostek.

- Selekcja elitarna (ang. *Elitist selection*)

Niekiedy dodatkowo do wyżej wymienionych metod, stosuje się selekcję elitarną, czyli wybór kilku najlepszych osobników, którzy w niezmienionej formie trafią bezpośrednio do następnej populacji.

W kolejnym kroku, osobniki znajdujące się we wcześniej wyselekcjonowanej grupie rodziców dobierane są w pary, a każda z nich poddawana jest procesowi krzyżowania. Odbiera się on poprzez zamianę wybranej części genów między genotypami rodziców, co skutkuje powstaniem dwóch nowych potomków. Krzyżowanie osobników wysoko ocenionych zwiększa szansę przekazania potomstwu najlepszych cech pochodzących od obydwu rodziców oraz w konsekwencji otrzymania lepiej przystosowanych jednostek. Niekiedy zdarzy się, że potomek otrzyma niekorzystne właściwości, zwłaszcza na początkowym etapie działania GA, kiedy genotypy zostają wygenerowane losowo, a prawdopodobieństwo wystąpienia wśród nich dobrych cech jest niewielkie. W takiej sytuacji, sam proces krzyżowania nie byłby wystarczający, ponieważ skutkowałby powielaniem tych samych wad genetycznych.

Rozwiązaniem tego problemu są mutacje, którym poddawany jest każdy z potomków, czyli losowe zmiany zachodzące w genotypie, które zwiększą różnorodność materiału genetycznego oraz okazjonalnie mogą zaowocować powstaniem dobrych cech, które pozytywnie wpłyną na ocenę danego osobnika i podniosą szansę przekazania jego genów kolejnym pokoleniom.

Zbiór wszystkich świeżo powstały osobników stanowi nową populację, nazywaną również pokoleniem, która zastępuje poprzednią. Wyżej opisany cykl powtarza się aż do uzyskania zadowalającego rozwiązania lub osiągnięcia maksymalnej liczby generacji.

Dokładny opis reprezentacji genotypu oraz operatorów genetycznych (krzyżowania i mutacji), zastosowanych na potrzeby pracy został przedstawiony w podrozdziale 4.2.

## **4. ŚRODOWISKO BADAWCZE**

W poprzednich rozdziałach przedstawiono opis wirtualnego środowiska na poziomie teoretycznym, a także omówiono ogólne zasady działania i różnice w wykorzystanych sieci neuronowych oraz algorytmu genetycznego. Niniejszy rozdział został poświęcony na szczegółowe omówienie tych elementów oraz dokładne przedstawienie zaimplementowanego środowiska.

Utworzone na potrzeby pracy, wirtualne środowisko badawcze jest dwuwymiarową przestrzenią o wymiarach 512x512 wm (wirtualnych metrów). W przeciwieństwie do mapy zamkniętej, wykorzystanej w pracach [15] i [16], w której agent po znalezieniu się na krawędzi środowiska nie może poruszać się dalej, w tym przypadku na wzór pracy [12], została zastosowana mapa toroidalna. Oznacza to, że agent poruszając się w takiej przestrzeni, po zbliżeniu się do jednej z krawędzi oraz przekroczeniu jej, zostaje przeniesiony na przeciwną stronę. Zmysły agentów zostały zaimplementowane w taki sposób, żeby były w stanie wykryć obiekty znajdujące się "poza granicami" mapy. Takie rozwiązanie imituje nieskończenie zapętloną przestrzeń oraz zapewnia ciągłość ruchu w taki sposób, że agenci nie zauważą, że w rzeczywistości żyją w ograniczonym świecie.

Zastosowanie toroidalnej przestrzeni ma na celu wykluczenie nagradzania osobników za gromadzenie się przy krańcach mapy, w szczególności w jej narożnikach, w których skuteczność ataków drapieżników jest znacznie niższa. To zachowanie nie wynika z decyzji podejmowanych przez agenta, lecz z ograniczeń wirtualnego środowiska, które nie występują w rzeczywistości. Przestrzeń toroidalna niweluje ten problem, dzięki wymuszeniu na osobnikach grupowania się w skutek ich własnej decyzji, a nie dotarcia do krańca dozwolonego obszaru.

W tak zaprojektowanej przestrzeni, na początku każdej symulacji umieszczane są agenci, które następnie, przez określona liczbę kroków żyją w nim i rywalizują między sobą. Dodatkowo, mogą również zostać wygenerowane obiekty roślin. Podczas trwania symulacji, zachowania agentów poddawane są ocenie, wykorzystywanej w dalszym etapie przez algorytm genetyczny do ewolucji agentów. Ofiary starają się przetrwać jak najdłużej i w jak największej liczności, natomiast drapieżniki próbują jak najszybciej upolować jak największą liczbę ofiar.

W każdym eksperymencie, środowisko badawcze jest zależnie od potrzeb odpowiednio dostosowywane, tak aby możliwa była weryfikacja postawionej w pracy tezy. Niektóre z mechanizmów i umiejętności agentów, takie jak komunikacja, energia życiowa, zdobywanie pożywienia i rośliny, mogą być w środowisku obecne lub też nie. Występowanie drapieżników oraz ich liczba również ulega zmianie, w przeciwieństwie do eksperymentów wykonywanych w pracach [12, 15, 16], w których brano pod uwagę wyłącznie jednego drapieżnika. Wszystko to ma na celu zbadanie wpływu poszczególnych czynników na przebieg tworzenia się zachowań stadnych.

## **4.1. AGENTY**

Każdy agent reprezentuje osobnika żyjącego w środowisku. Może on być ofiarą lub drapieżnikiem. Nie posiada fizycznych rozmiarów i istnieje jako punkt w przestrzeni dwuwymiarowej, posiadający określoną pozycję (x i y), kąt obrotu (aktualny kierunek w którym jest zwrócony), wektor obserwacji oraz parametry charakteryzujące osobnika z danego gatunku.

Wirtualne organizmy posiadają indywidualne cechy, które wynikają z rzeczywistych różnic obserwowanych w naturze między gatunkami ofiar oraz drapieżników. Wszystkie parametry agentów dotyczące poruszania się oraz zmysłów, zostały ustalone na wzór tych wykorzystanych w pracach [12, 15, 16], co ma na celu łatwiejsze porównanie uzyskanych wyników.

### **4.1.1. AGENT OFIARY**

Ofiary poruszają się trzy razy wolniej od drapieżnika, lecz wykazują większą zwinność, dzięki wyższej prędkości skrętu. Agent ofiary podejmuje decyzje dotyczące poruszania się w przód / tył oraz skrętu w lewo / prawo.

Ponadto dysponują umiejętnością komunikacji, która sprowadza się do automatycznego wydania jednego z dwóch rodzajów dźwięków. Następuje to zależnie od sytuacji, kiedy w polu widzenia ofiary znajdzie się pożywienie w postaci rośliny lub drapieżnik. Wykształcenie zdolności wydawania dźwięków w takich sytuacjach drogą ewolucji, mogłoby zająć wiele czasu, dlatego zdecydowano się ją zaimplementować i z góry narzucić znaczenie konkretnych sygnałów, jednak ich poprawna interpretacja oraz reakcja na odebranie któregoś z nich zależy wyłącznie od agenta.

Organizmy ofiar posiadają również energię życiową, która maleje wraz z upływającym czasem. Kiedy spadnie do zera, dany osobnik ginie i zostaje usunięty z symulacji. W celu uzupełnienia zasobów energii, ofiary muszą zdobywać pożywienie występujące w postaci roślin. Kiedy w polu widzenia poruszającej się w środowisku ofiary, znajdzie się roślina i będzie ona dostatecznie blisko agenta, zostanie automatycznie skonsumowana. Agent może zjeść roślinę co pewną określoną liczbę kroków, co ma obrazować czas potrzebny na konsumpcję pożywienia w rzeczywistości.

### **4.1.2. AGENT DRAPIEŻNIKA**

Drapieżników charakteryzuje trzykrotnie większa prędkość poruszania się w stosunku do ofiar, jednakże wiąże się ze zmniejszoną szybkością skrętu (mniejszą zwinnością). Posiadają również większy zasięg zmysłu wzroku niż ofiary. Agent drapieżnika podejmuje decyzje dotyczące poruszania się w przód / tył oraz skrętu w lewo / prawo, ale dodatkowo może zdecydować się na wydawanie dźwięku. Analogicznie jak w pracach [12, 15, 16], drapieżnicy żyjący w środowisku są nieśmiertelní i w przeciwnieństwie do ofiar nie mogą zostać upolowani czy zginąć z głodu (braku energii).

W celu zbadania wpływu konieczności zdobywania pożywienia przez ofiary, zaimplementowano mechanizm energii życiowej, lecz wyłącznie w przypadku ofiar. Zdecydowano się na taką rozbieżność między gatunkami, ze względu na różnice w ich funkcji oceny. Drapieżnik jest nagradzany bezpośrednio za liczbę upolowań, w związku z czym wprowadzenie dodatkowej kary dla osobników słabo polujących, w postaci śmierci spowodowanej brakiem energii, nie miałoby istotnego znaczenia. Ofiary nie są nagradzane za zdobywanie pożywienia, lecz przetrwanie w jak największej liczności, dlatego w ich przypadku, konieczność znalezienia pokarmu będzie wyłącznie pośrednim zadaniem w dążeniu do celu wyznaczonego funkcją oceny. Uznano, że możliwość śmierci głodowej drapieżników wprowadziłaby jedynie zbędne urozmaicenie, nieistotne dla rodzaju badań przeprowadzanych w niniejszej pracy.

Główym zadaniem drapieżników żyjących w środowisku jest polowanie na ofiary. Drapieżnik przemieszcza się w poszukiwaniu potencjalnego celu, a w chwili, w której cel pojawi się w polu widzenia oraz znajdzie się odpowiednio blisko, drapieżnik podejmuje próbę ataku.

Ataki drapieżników są obarczone wizyjnym ograniczeniem w postaci mechanizmu dezorientacji, który jest jednym z kluczowych czynników prowadzących do wykształcenia się mechanizmów grupowania wśród zwierząt (w stada, roje, ławice), co zostało zaobserwowane w środowisku naturalnym, a w przypadku sztucznego życia potwierdzone w pracy [15]. Kiedy jest aktywny, szanse drapieżnika na przeprowadzenie skutecznego ataku i schwytanie docelowej ofiary maleją, gdy w polu widzenia drapieżnika widoczne są jakiekolwiek inne ofiary, znajdujące się w pobliżu ofiary obranej za cel. To percepcyjne ograniczenie jest wzorowane na obserwacjach drapieżników polujących w naturze, których skuteczność, czyli stosunek udanych ataków do ogólnej liczby prób ataków, jest znacznie mniejsza, kiedy celem staje się stado ofiar o większej gęstości.

$$P_{capture} = \frac{k}{N_{near}} \quad (4.1)$$

Prawdopodobieństwo udanego ataku  $P_{capture}$ , określa wzór (4.1), gdzie  $k$  to współczynnik dezorientacji (tutaj wynosi 1), natomiast  $N_{near}$  to liczba agentów ofiar, które znajdują się w polu widzenia drapieżnika i w określonej odległości od ofiary będącej celem ataku (w sąsiedztwie celu). Według powyższego wzoru, drapieżnik ma 50% szans na złapanie celu z jedną widoczną ofiarą w jego sąsiedztwie, 33% szans na powodzenie z dwiema ofiarami w sąsiedztwie celu, itd. W związku z tym, ofiary są w stanie wykorzystać efekt dezorientacji na swoją korzyść, dzięki formowaniu stad.

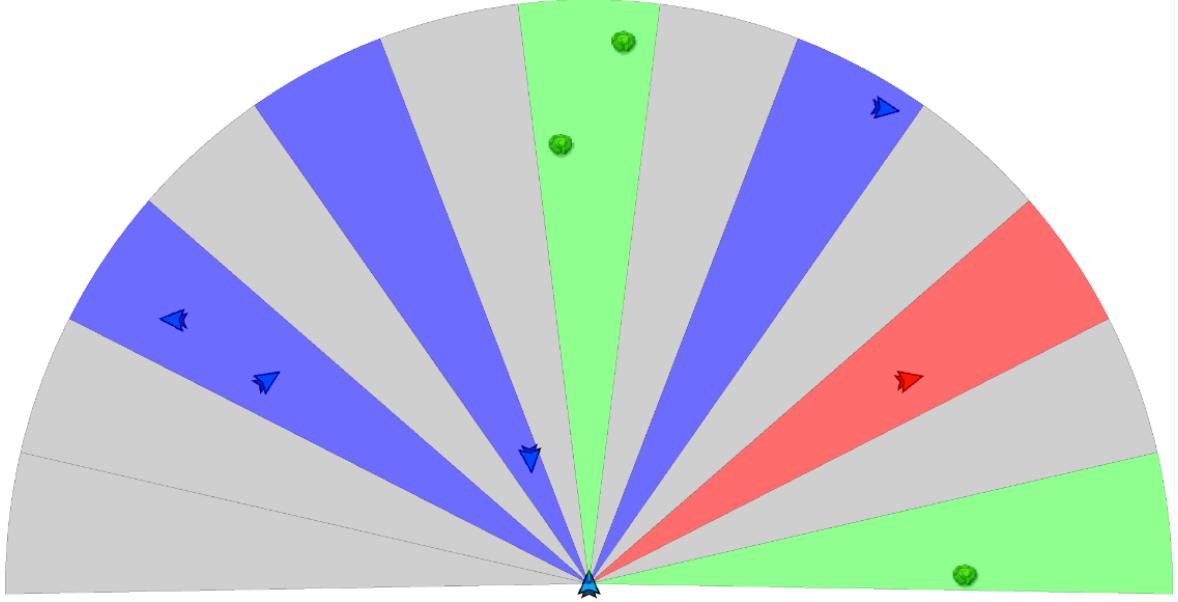
Drapieżnik może przeprowadzić próbę ataku tylko raz na kilka kroków symulacji, co ma obrazować konsumpcję ofiary po udanym ataku, potrzebę regeneracji sił (odpoczynku), czy konieczność zorientowania się w sytuacji w przypadku zgubienia lub miniecia swojego celu (nieudanej próby ataku).

Drapieżnicy dysponują również umiejętnością komunikacji, która ogranicza się wyłącznie do jednego rodzaju dźwięku. W przeciwieństwie do ofiar, sami decydują kiedy wydać sygnał, w związku z czym nie ma on odgórnie przypisanego znaczenia. Zadaniem drapieżników jest samodzielne nadanie mu interpretacji oraz znalezienie sposobu wykorzystania.

Samotnie polujący drapieżnik, ze względu na mechanizm dezorientacji, ma niewielkie szanse przeprowadzenia skutecznego ataku na grupę ofiar. Umiejętność komunikacji może być wykorzystana do zorganizowania wspólnych ataków, dzięki którym szanse powodzenia wzrosną proporcjonalnie do liczby drapieżników polujących na daną grupę, a dodatkowo, wraz ze zmniejszającą się licznością stada, szanse na złapanie kolejnych osobników rosną. Istnieje również możliwość, że stado przy obecności większej liczby drapieżników ulegnie rozproszeniu, dzięki czemu możliwe będą mogli ataki na pojedyncze, oddalające się jednostki. Takie zachowania obserwowane są w naturze.

#### 4.1.3. ZMYSŁY AGENTÓW

Agenty wyposażone są w dwa rodzaje zmysłów - wzrok oraz słuch, za pomocą których odbierają bodźce pochodzące ze środowiska, umożliwiające interakcje z otoczeniem. Wzrok jest podstawowym zmysłem, który stanowi główne źródło informacji pochodzących ze środowiska. Pozwala na dostrzeżenie, ocenę odległości oraz rozróżnienie pobliskich obiektów, takich jak ofiary, drapieżnicy, czy rośliny. Został on zobrazowany na rysunku 4.1.



Rys. 4.1. Schemat poglądowy przedstawiający pole wzroku agenta. Kolorami zostały zaznaczone wycinki pola widzenia (segmenty), w których znajdują się poszczególne obiekty. Niebieski wycinek koła zawiera agenta ofiary, czerwony agenta drapieżnika, natomiast zielony obiekt roślinę.

Wzrok agenta ma określony zasięg oraz kąt, wynoszące odpowiednio 100 cm (w przypadku drapieżnika 200 cm) oraz  $182^\circ$ . Pole wzrokowe dzieli się na 13 segmentów, z których każdy potrafi wykryć, czy jakiś obiekt znajduje się w obejmowanym przez niego obszarze. Jeśli tak, to według wzoru (4.2) przypisywana jest mu wartość  $v$  z przedziału  $\langle 0, 1 \rangle$ , równa odległość obiektu od agenta  $d_{object}$ , znormalizowanej względem zasięgu widzenia  $r_{vision}$ .

$$v = 1 - \frac{d_{object}}{r_{vision}} \quad (4.2)$$

W skrajnych przypadkach  $v$  przyjmuje wartość 1, kiedy obiekt jest tuż przez agentem oraz 0, gdy znajdzie się poza zasięgiem. Jeśli w danym segmencie pola widzenia znajdzie się więcej niż jeden obiekt, pod uwagę brany jest wyłącznie najbliższy. Każdy segment dodatkowo dzieli się na trzy warstwy, odpowiedzialne za wykrywanie odpowiednio ofiar, drapieżników lub roślin. W niektórych badaniach nie występuje obiekt pożywienia, dlatego w takim przypadku wzrok posiada jedynie dwie warstwy: ofiar i drapieżników. Przykładowy wektor wartości przypisanych poszczególnym segmentom dla każdej z warstw, odpowiadający sytuacji poglądowej zaprezentowanej na rysunku 4.1, przedstawia tabela 4.1.

Tabela 4.1. Wartości wektora obserwacji wzrokowych dla sytuacji przedstawionej na Rys. 4.1

<b>Warstwa</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>
Ofiar	0	0	0.34	0	0.76	0	0	0	0.04	0	0	0	0
Drapieżników	0	0	0	0	0	0	0	0	0	0	0.34	0	0
Roślin	0	0	0	0	0	0	0.24	0	0	0	0	0	0.35

Zmysł słuchu działa w podobny sposób, umożliwiając organizmom odbieranie pochodzących ze środowiska sygnałów dźwiękowych. Zasięg jego działania dla danego rodzaju agenta wynosi tyle co zasięg wzroku, lecz jego kąt jest równy  $360^\circ$ , obejmując obszar wokół agenta. Obszar słuchu został podzielony na 12 segmentów, z których każdy jest opisany ciągiem bitów o długości

odpowiadającej liczbie różnych dźwięków możliwych do wydania przez agenta. Wartość 1 na pojedynczym bicie świadczy o tym, że z kierunku, za który odpowiada dany segment doszedł pewien dźwięk. W przeciwnym wypadku bit przyjmuje wartość 0. Przykładowy wektor pobudzeń zmysłu słuchu dla agenta ofiary został przedstawiony w tabeli 4.2. Wektor dla drapieżnika wygląda analogicznie, lecz występuje w nim wyłącznie jeden rodzaj dźwięku.

Tabela 4.2. Przykładowe wartości wektora obserwacji słuchowych agenta ofiary.  
Symbole 'a' oraz 'b' oznaczają dwa rodzaje dźwięków.

1	2	3	4	5	6	7	8	9	10	11	12
a	b	a	b	a	b	a	b	a	b	a	b
1	0	0	0	0	1	0	0	0	1	0	0

#### 4.1.4. STEROWNIK AGENTÓW

Każdy z agentów posiada własny sterownik, pełniący rolę "sztucznego mózgu", którego zadaniem jest podejmowanie decyzji dotyczącej następnego ruchu. Na wejściu otrzymuje on wektor obserwacji  $O$ , który składa się z bodźców odebranych ze środowiska za pomocą zmysłu wzroku oraz słuchu. Wyjściem sterownika jest wektor akcji  $A$ , którego wartości określają prędkość i kierunek poruszania się agenta, a w przypadku drapieżnika również decyzję wydania dźwięku. Schemat budowy sterownika został przedstawiony na rysunku 4.2.

Głównym elementem sterownika jest sieć rekurencyjna - LSTM lub RNN, które zostały szczegółowo opisane w rozdziale 3. Różnica między nimi tkwi w wewnętrznej strukturze oraz procesie przetwarzania danych, jednak sposób ich wykorzystania wygląda tak samo. Na wejściu otrzymuję wektor  $O$ , a zwracany jest wektor wyjściowy  $W$ . Rozmiar wektora  $O$  określa poniższy wzór (4.3):

$$N_{obs} = V_{cells} \cdot V_{layers} + H_{cells} \cdot H_{sounds} \quad (4.3)$$

gdzie  $V_{segments}$  oraz  $V_{layers}$  są odpowiednio liczbą segmentów i warstw zmysłu wzroku, natomiast  $H_{segments}$  oraz  $H_{sounds}$  są liczbą segmentów zmysłu słuchu i dźwięków możliwych do wydania. Jak wynika z powyższego wzoru, wartość  $N_{obs}$  nie jest stała i zależy od gatunku agenta (ofiary lub drapieżnika), czy obecności zmysłu słuchu. Wektor wyjściowy  $W$ , otrzymany z sieci rekurencyjnej trafia następnie do pojedynczej warstwy neuronów, która według wzoru 3.4, przekształca go do wektora akcji  $A$ . Jej funkcją aktywacji jest tangens hiperboliczny, dlatego wartości wektora  $A$ , mieszczą się w przedziale  $(-1, 1)$ .

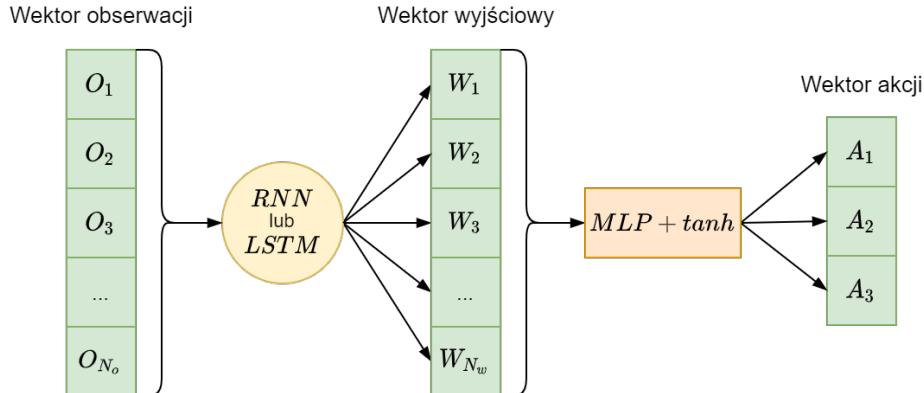
Tabela 4.3. Akcje agentów zależnie od wartości  $A_i$

Wartość	$A_1$	$A_2$	$A_3$ (drapieżnik)
$A_i > 0$	ruch w przód z szybkością $v_{move}$	skręt w lewo z szybkością $v_{turn}$	wydanie dźwięku
$A_i < 0$	ruch w tył z szybkością $v_{move}$	skręt w prawo z szybkością $v_{turn}$	
$A_i = 0$	brak ruchu	brak skrętu	brak dźwięku

$$v_{move} = abs(A_1) \cdot v_{move\_max} \quad v_{turn} = abs(A_2) \cdot v_{turn\_max}$$

Tabela 4.3 przedstawia akcje podejmowane przez agenta, zależnie od poszczególnych wartości wektora  $A$ . Dwie pierwsze z nich ( $A_1$  i  $A_2$ ), są wspólne dla obydwu gatunków i odpowiadają za poruszanie się oraz skręt. Ich wartość bezwzględna decyduje o szybkości, natomiast znak o kierunku. Maksymalna szybkość ruchu i skrętu jest osiągana dla  $A_1 = A_2 = 1$  i określana przez parametry  $v_{move\_max}$  i  $v_{turn\_max}$ , których wartości zależą od gatunku danego agenta. Trzecia

wartość  $A_3$ , jest odpowiedzialna za wydawanie dźwięku w danym kroku czasowym i występuje tylko w przypadku agenta drapieżnika oraz przy włączonym mechanizmie komunikacji. Wszystkie wartości wektora  $A$  są od siebie niezależne. Agent może równocześnie poruszać się, wykonać skręt oraz wydać dźwięk (w przypadku drapieżnika).



Gdzie:

- $O$  - wektor obserwacji
- $N_o$  - rozmiar wektora obserwacji
- $W$  - wektor wyjściowy sieci rekurencyjnej
- $N_w$  - rozmiar wektora wyjściowego
- $RNN/LSTM$  - sieć rekurencyjna RNN lub LSTM
- $MLP + tanh$  - dodatkowa warstwa sieci neuronowej z funkcją aktywacji tangens hiperboliczny
- $A$  - wektor akcji agenta
- $A_1$  - zmienna prędkości ruchu
- $A_2$  - zmienna prędkości skrętu
- $A_3$  - zmienna wydawania dźwięku  
(tylko dla drapieżnika)

Rys. 4.2. Schemat budowy sterownika

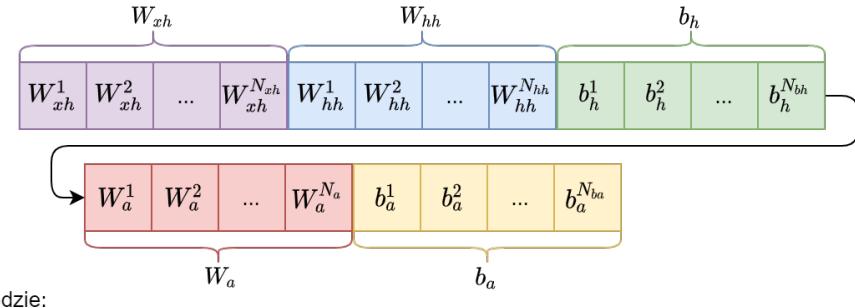
## 4.2. ALGORYTM GENETYCZNY

Algorytm genetyczny jest mechanizmem odpowiedzialnym za proces ewolucji zachodzący w symulacji, który zapewnia agentom możliwość adaptacji do warunków panujących w środowisku. Ogólny opis jego działania został przedstawiony w podrozdziale 3.4, natomiast pozostałe kwestie związane z budową genotypu oraz zastosowanymi operatorami genetycznymi, zostały opisane w dalszej części tego rozdziału.

### 4.2.1. GENOTYP

Genotyp jest jednowymiarową tablicą wartości, w której zakodowane są wszystkie parametry wymagane do poprawnego działania sterownika agentów. Powstaje po spłaszczeniu wszystkich warstw neuronów wchodzących w skład sterownika oraz umieszczeniu ich kolejno w jednym, długim wektorze. Proces ten można odwrócić poprzez wycięcie odpowiednich fragmentów genotypu oraz przekształcenie w odpowiadające im dwuwymiarowe macierze (warstwy). Na rysunkach 4.3 oraz 4.4 przedstawiono schemat genotypu sterownika wykorzystującego sieć RNN oraz LSTM.

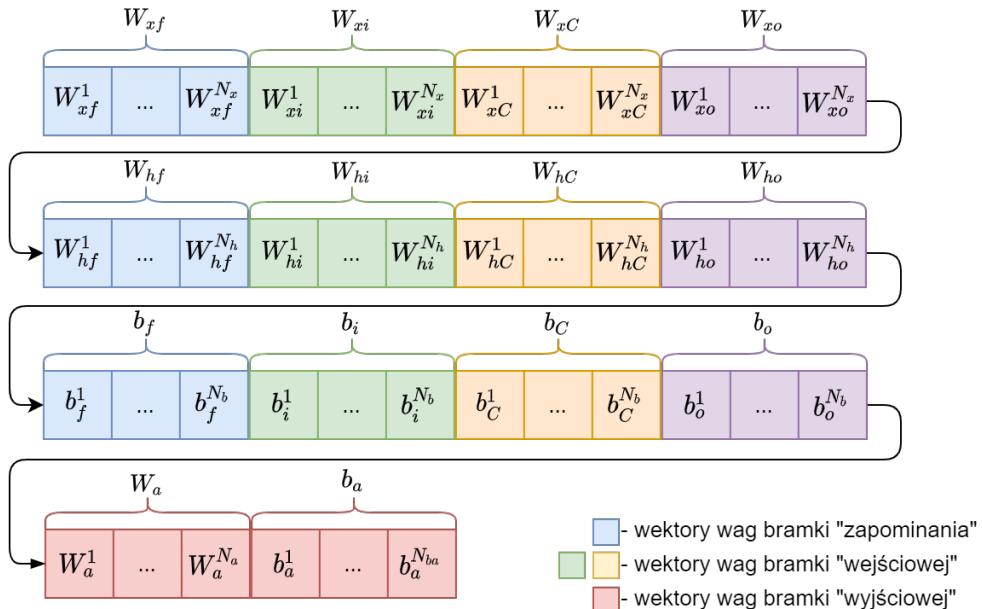
Nowy sterownik jest zawsze inicjowany w formie genotypu, w którym z pewnym prawdopodobieństwem  $P_{init}$ , części z genów przypisywana jest wartość pochodząca z rozkładu normalnego  $\mathcal{N}(\mu, \sigma^2)$ , natomiast pozostałym wartość równa zero.



Gdzie:

- $W_{xh}$  - wektor wag warstwy stanu wejściowego
- $N_{xh}$  - rozmiar wektora  $W_{xh}$
- $W_{hh}$  - wektor wag warstwy stanu wewnętrznego
- $N_{hh}$  - rozmiar wektora  $W_{hh}$
- $W_a$  - wektor wag warstwy wyjściowej
- $N_a$  - rozmiar wektora  $W_a$
- $b_h$  - wektor biasów sieci rekurencyjnej
- $N_{bh}$  - rozmiar wektora  $b_h$
- $b_a$  - wektor biasów warstwy wyjściowej
- $N_{ba}$  - rozmiar wektora  $b_a$
- $N_{obs} = N_{obs} \cdot N_H$
- $N_{xh} = N_H \cdot N_H$
- $N_a = N_H \cdot N_A$
- $N_{bh} = N_H$
- $N_{ba} = N_A$
- $N_H$  - liczba neuronów ukrytych
- $N_{obs}$  - rozmiar wektora obserwacji
- $N_A$  - rozmiar wektora akcji

Rys. 4.3. Schemat genotypu sterownika RNN. Oznaczenia warstw neuronów pokrywają się z wykorzystanymi do opisu sieci RNN w rozdziale 3.2.



- wektory wag bramki "zapominania"
- wektory wag bramki "wejściowej"
- wektory wag bramki "wyjściowej"

Gdzie:

- $W_{xf}, W_{xi}, W_{xC}, W_{xo}$  - wektory wag warstw stanu wejściowego dla każdej bramki
- $W_{hf}, W_{hi}, W_{hC}, W_{ho}$  - wektory wag warstw stanu wewnętrznego dla każdej bramki
- $b_f, b_i, b_C, b_o$  - wektory biasów warstw dla każdej bramki
- $N_x$  - rozmiar każdego z wektorów  $W_{xf}, W_{xi}, W_{xC}, W_{xo}$
- $N_h$  - rozmiar każdego z wektorów  $W_{hf}, W_{hi}, W_{hC}, W_{ho}$
- $N_b$  - rozmiar każdego z wektorów  $b_f, b_i, b_C, b_o$
- $W_a, b_a$  - wektor wag i biasów warstwy wyjściowej
- $N_a$  - rozmiar wektora  $W_a$
- $N_{ba}$  - rozmiar wektora  $b_a$
- $N_x = N_{obs} \cdot N_H$
- $N_h = N_H \cdot N_H$
- $N_b = N_H$
- $N_a = N_H \cdot N_A$
- $N_{ba} = N_A$
- $N_H$  - liczba neuronów ukrytych
- $N_{obs}$  - rozmiar wektora obserwacji
- $N_A$  - rozmiar wektora akcji

Rys. 4.4. Schemat genotypu sterownika LSTM. Oznaczenia warstw neuronów pokrywają się z wykorzystanymi do opisu sieci LSTM w rozdziale 3.3.

#### 4.2.2. OPERATORY EWOLUCYJNE

Zadaniem operatorów ewolucyjnych jest modyfikacja genotypów bieżącego pokolenia w taki sposób, aby przekazać kolejnemu pokoleniu najlepsze cechy oraz umożliwić powstawanie nowych, które przyczynią się do powstania potencjalnie lepszych osobników niż obecne. W pracy przetestowane zostały różne metody oraz wartości parametrów algorytmu genetycznego, w celu znalezienia kombinacji, dzięki którym proces ewolucji zachowań stadnych będzie jak najefektywniejszy. Badania dotyczące doboru parametrów uczenia znajdują się w rozdziale 5.1.

Do utworzenia potomków oraz przekazania im cech odziedziczonych po rodzicach stosuje się krzyżowanie. W pracy przetestowano jego następujące warianty:

- Krzyżowanie jedno-punktowe  
Losowany jest jeden punkt w genotypie, a następnie wszystkie geny znajdujące się przed nim, są zamienianie między genotypami rodziców.
- Krzyżowanie dwu-punktowe  
Losowane są dwa punkty w genotypie, a następnie tak wyznaczony segment ulega zamianie pomiędzy genotypami rodziców.
- Krzyżowanie jedno-punktowe na każdy segment  
Przebiega analogiczne jak krzyżowanie "jedno-punktowe", przy czym nie jest ono wykonywane na całości genotypu, lecz jego fragmentach, odpowiadających poszczególnym warstwom sterownika. Te fragmenty przedstawione zostały na rysunku 4.2.1 i 4.2.1.
- Krzyżowanie dwu-punktowe na każdy segment  
Przebiega analogiczne jak krzyżowanie "jedno-punktowe na każdy segment", przy czym na każdy fragment genotypu wykonywane jest krzyżowanie "dwu-punktowe".

W celu zapewnienia zróżnicowania genetycznego oraz wprowadzenia zmian w genotypie stosuje się metody mutacji. W pracy przetestowano jej następujące odmiany:

- Mutacja genu  
Każdy gen (pojedyncza wartość genotypu), ma szansę z prawdopodobieństwem  $P_{mut}$  ulec mutacji, polegającej na przypisaniu mu nowej, losowej wartości pochodzącej z rozkładu normalnego  $\mathcal{N}(\mu, \sigma^2)$ .
- Wypełnienie genów  
Zachodzi z prawdopodobieństwem  $P_{fill}$ . Losowane są dwa punkty w genotypie, a wszystkie geny znajdujące się pomiędzy nimi są usuwane. Następnie, w ich miejsce losowane są zupełnie nowe wartości pochodzące z rozkładu  $\mathcal{N}(\mu, \sigma^2)$ .
- Powielenie genów  
Zachodzi z prawdopodobieństwem  $P_{dup}$ . Losowane są dwa punkty w genotypie, a następnie wszystkie wartości genów znajdujące się pomiędzy nimi są duplikowane w nowe miejsce.
- Usunięcie genów  
Zachodzi z prawdopodobieństwem  $P_{del}$ . Losowane są dwa punkty w genotypie, a wszystkie geny znajdujące się pomiędzy nimi są usuwane. Następnie, na ich miejsce przesuwane są wartości kolejnych genów (następuje scalenie obydwu stron genotypu). Na koniec, brakujące geny uzupełniane są losowymi wartościami z rozkładu  $\mathcal{N}(\mu, \sigma^2)$ .

#### 4.2.3. FUNKCJA PRZYSTOSOWANIA

Każdy osobnik z populacji zostaje umieszczony w symulacji oraz ewaluowany, w celu uzyskania oceny skuteczności jego przystosowania do życia w środowisku. Dzięki temu, algorytm genetyczny jest w stanie dokonać selekcji najbardziej obiecujących osobników z każdej generacji oraz na ich podstawie podjąć próbę stworzenia nowego, lepszego pokolenia. W pracy zastosowano dwie funkcje przystosowania, ponieważ cele ofiar oraz drapieżników żyjących w środowisku różnią się. Ofiary dążą do tego, aby ich stado przetrwało jak najdłużej oraz w jak największej liczności, natomiast drapieżnicy skupią się na jak najsukutecznieszym polowaniu na ofiary.

Funkcja przystosowania ofiary:

$$F_{prey} = \sum_{t=1}^N Alive_t \quad (4.4)$$

Funkcja przystosowania drapieżnika:

$$F_{predator} = \sum_{t=1}^N (N_{prey} - Alive_t) \quad (4.5)$$

gdzie  $t$  jest bieżącym krokiem symulacji,  $N$  liczbą wszystkich kroków w symulacji,  $N_{prey}$  początkową wielkością populacji agentów ofiar, a  $Alive_t$  liczbą żywych agentów ofiar po kroku  $t$ .  $N_{prey} - Alive_t$  odpowiada liczbie udanych ataków drapieżników po kroku  $t$ .

Można zauważyć, że funkcje przystosowania obydwu agentów są od siebie zależne. Im lepiej przystosowane ofiary (większa przeżywalność), tym niższą ocenę otrzyma drapieżnik (upolował mniej ofiar). Ta zależność działa również w drugą stronę. Ponadto, ocena skuteczności wszystkich ofiar oraz drapieżników żyjących w środowisku jest wspólna dla osobników danego gatunku.

Zastosowanie takich funkcji przystosowania możliwe było dzięki temu, że w każdej symulacji ewaluowany jest wyłącznie jeden genotyp ofiary oraz drapieżnika, natomiast każdy osobnik występujący w środowisku jest jego klonem. Wstępnie przeprowadzone badania pokazały, że takie podejście zwiększa jakość stad oraz przyspiesza sam proces ich powstawania. W przypadku, w którym każdy osobnik w środowisku posiadał osobny genotyp oraz sterownik, a tym samym własną funkcję przystosowania, ewolucja zachowań stadnych wśród agentów ofiar przebiegała znacznie trudniej oraz była możliwa wyłącznie przy odpowiednio dobranych parametrach środowiska.

### 4.3. SYMULACJA

Badania eksperymentalne przeprowadzane na potrzeby pracy, odbywają się z wykorzystaniem symulacji, która rozpoczyna się od wygenerowania losowych populacji  $N_{prey}$  ofiar oraz  $N_{predator}$  drapieżników. Zależnie od badania, do każdego agenta może zostać przypisany nowy lub już wcześniej wyuczony genotyp.

Następnie, bez zwracania losowane są pary ofiara-drapieżnik. Dla każdej z nich, agenci są klonowani k-krotnie (zależnie od wartości parametru  $N_{prey}$  oraz  $N_{predator}$ ), a każdy z klonów umieszczany jest w środowisku. Następuje ich ewaluacja w celu uzyskania wartości funkcji przystosowania. Po wykonaniu tego dla każdej pary, z wykorzystaniem algorytmu genetycznego tworzone jest nowe pokolenie agentów. Takie kroki kończą jedną generację, a następnie są powtarzane (z wyjątkiem generowania populacji, ponieważ wykorzystywana jest bieżąca). Cała symulacja kończy się po wykonaniu  $N_{generations}$  generacji.

Ewaluacja osobników w wirtualnym środowisku rozpoczyna się od wygenerowania agentów oraz rozmieszczenia ich w losowych miejscach na mapie. Następnie, przez  $N_{steps}$  iteracji, ich zachowania poddawane są ocenie, która wykorzystana jest jako wartość fitness. Pojedyncza iteracja rozpoczyna się od aktualizacji bodźców wzrokowych i słuchowych oraz wyliczenia wektora obserwacji dla każdego agenta. Na ich podstawie, sterownik podejmuje decyzje dotyczące następnego ruchu. W tym momencie wyliczane są nowe pozycje osobników oraz zależnie od sytuacji, wykonywane próby ataków drapieżników i zdobycia pożywienia przez ofiary, a także wydawane są stosowne dźwięki. Na koniec iteracji następuje aktualizacja oceny agentów oraz miar wykorzystywanych do porównania wyników badań.

Podejście z tworzeniem klonów agenta oraz przypisywanie im wspólnej oceny zostało zastosowane również w pracach [12, 15, 16]. Taki sposób generowania stada ofiar ma na celu wyeliminowanie efektu "samolubnego stada" na poziomie pojedynczej jednostki. Kloni posiadają wspólną ocenę, dlatego ważna jest ich ogólna sprawność, a nie pojedynczych jednostek. Samolubne zachowanie poszczególnych osobników negatywnie wpłynie na wspólną ocenę stada. Ponieważ wszystkie ofiary są identyczne (posiadają te same geny i ten sam sterownik), metoda selekcji oraz rozmieszczenia ich w środowisku, również nie mają wpływu na jakość symulacji. Dobrze przystosowany osobnik nie ma przewagi w przypadku wylosowania osobników słabych do reszty stada lub w wyniku początkowego umieszczenia go w bardziej sprzyjającym położeniu.

Wyżej opisaną symulację, zapisaną w postaci pseudokodu, przedstawia algorytm 1. Szczegóły implementacyjne zostały tutaj pominięte, ponieważ celem było pokazanie kolejności i ogólnego schematu działania symulacji.

Wszystkie parametry środowiska zostały przedstawione w tabeli 4.4. Część z nich jest niezmienna na przestrzeni wszystkich eksperymentów, lecz niektóre parametry są zależne od wymagań poszczególnych badań, dlatego nie zostały sprecyzowane, np. włączony mechanizm komunikacji czy obecność pożywienia ofiar. Stosowne informacje zostały zamieszczone przy każdym eksperymencie.

---

**Algorithm 1:** Przebieg symulacji

---

```
InitializePopulations
repeat NumberOfGenerations times
    InitializeEnvironment
    repeat NumberOfSimulationSteps times
        foreach agent do
            UpdateObservations
            foreach agent do
                if CanAttack / CanEat then
                    TryAttack / TryEat
            foreach agent do
                UpdateDecisions
            foreach agent do
                UpdateMovements
            UpdateMeasures
            UpdateFitness
    NextGeneration
```

*InitializePopulations*: wygenerowanie początkowej populacji agentów ofiar oraz drapieżników lub wczytanie już wcześniej wyuczonych genotypów.

*InitializeEnvironment*:inicjalizacja środowiska symulacyjnego w sposób odpowiedni dla danego badania, czyli wygenerowanie agentów i obiektów oraz losowe umieszczenie ich w środowisku.

*UpdateObservations*: aktualizacja bodźców wzrokowych oraz słuchowych, obliczenie wartości wektora wejściowego (obserwacji) agenta.

*TryAttack / TryEat*: kiedy zostaną spełnione odpowiednie warunki, zostanie przeprowadzona próba ataku drapieżnika lub zdobycie pożywienia przez ofiarę.

*UpdateDecisions*: aktualizacja decyzji, obliczenie wartości wektora akcji agenta na podstawie obserwacji (wektora wejściowego) oraz stanów wewnętrznych sieci rekurencyjnych - RNN lub LSTM, zależnie od rodzaju sterownika.

*UpdateMovements*: wykonanie akcji wynikających z wektora wyjściowego (podjętych przez agenta w kroku podejmowania decyzji).

*UpdateMeasures*: aktualizacja miar niezbędnych do porównania wyników. Ich opis znajduje się w rozdziale 5.

*UpdateFitness*: aktualizacja wartości funkcji przystosowania agentów na podstawie wzoru 4.4 oraz 4.5.

*NextGeneration*: na podstawie ustalonych wartości funkcji przystosowania poszczególnych osobników, algorytm genetyczny tworzy nowe pokolenie agentów.

---

Tabela 4.4. Wartości parametrów środowiska.

Nazwa parametru	Wartość
<i>Parametry symulacji</i>	
Rozmiar mapy	512 wm x 512 wm
Liczba generacji	1000
Liczba kroków symulacji	2000
Początkowa liczba kroków bez drapieżnika	250
Metoda rozmieszczenia roślin	<i>parametr dostrajany</i>
<i>Parametry agenta ofiary</i>	
Liczba kopii agenta na początku symulacji	50
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	<i>parametr dostrajany</i>
Maksymalna prędkość ruchu	1 wm / krok
Maksymalna prędkość skrętu	8° / krok
Zasięg wzroku	100 wm
Kąt wzroku	182°
Liczba segmentów wzroku	13
Kąt segmentu wzroku	14°
Zasięg słuchu	100 wm
Kąt słuchu	360°
Liczba segmentów słuchu	12
Kąt segmentu słuchu	30°
Początkowy zasób energii życiowej	100
Zużycie energii życiowej	0.2 / krok
Wartość energetyczna rośliny	50
Minimalna liczba kroków między kolejną konsumpcją	40
<i>Parametry agenta drapieżnika</i>	
Liczba kopii agenta na początku symulacji	<i>parametr dostrajany</i>
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	<i>parametr dostrajany</i>
Maksymalna prędkość ruchu	3 wm / krok
Maksymalna prędkość skrętu	6° / krok
Zasięg wzroku	200 wm
Kąt wzroku	182°
Liczba segmentów wzroku	13
Kąt segmentu wzroku	14°
Zasięg słuchu	100 wm
Kąt słuchu	360°
Liczba segmentów słuchu	12
Kąt segmentu słuchu	30°
Minimalna liczba kroków między kolejnymi atakami	10
Współczynnik dezorientacji	1
Promień bezpieczeństwa ofiary w mechanizmie dezorientacji	30 wm

## 5. BADANIA EKSPERYMENTALNE

Głównym celem eksperymentów jest zbadanie wpływu mechanizmu komunikacji występującego u agentów obydwu gatunków, a w przypadku ofiar dodatkowo konieczności zdobywania pożywienia, na formowanie się zachowań stadnych oraz strategii ataków drapieżników. Ponadto przeprowadzone zostanie porównanie wyników osiąganych w zależności od rodzaju zastosowanego sterownika - RNN lub LSTM.

Do porównania wyników eksperymentów niezbędne są miary, które pozwolą ocenić skuteczność zachowań powstających w określonych warunkach, pośród agentów ofiar oraz drapieżników żyjących w wirtualnym środowisku.

Miary wykorzystane podczas badań:

- **Przeżywalność** (ang. *Survivorship*)

Średnia liczba żywych agentów ofiar pod koniec symulacji.

- **Gęstość stada** (ang. *Swarm density*)

Średnia liczba agentów znajdująca się w promieniu 30 cm od danego organizmu, liczona w każdym kroku symulacji.

- **Rozpiętość stada** (ang. *Swarm dispersion*)

Średnia odległość do najbliższego osobnika tego samego gatunku, liczona w każdym kroku symulacji.

- **Liczba zdobytego pożywienia** (ang. *Number of food collected*)

Średnia liczba zdobytego pożywienia, przypadająca na pojedynczą ofiarę pożywienia pod koniec symulacji.

- **Liczba ataków** (ang. *Number of attacks*)

Średnia liczba prób (niezależnie od sukcesu) ataków, przypadająca na jednego drapieżnika pod koniec symulacji.

- **Liczba udanych ataków** (ang. *Number of successful attacks*)

Średnia liczba udanych ataków, przypadająca na jednego drapieżnika pod koniec symulacji.

W części z przeprowadzonych eksperymentów, zależnie od potrzeb, zastosowano już wcześniej wyuczone genotypy agentów (ofiar lub drapieżników), które zostały również wyłączone z procesu dalszej ewolucji. Ma to na celu przyspieszenie przebiegu symulacji, dzięki pominięciu kształtowania się podstawowych cech wymaganych do przeprowadzenia danego eksperymentu. W dodatku, wykorzystanie genów wyłącznie najlepszego osobnika (z wcześniej wyuczonej populacji), wyeliminowało efekt losowy polegający na doborze słabo przystosowanej jednostki, co mogłoby spowodować niepożądane zawyżenie wyników. Szczegółowe informacje zostały zamieszczone przy każdym eksperymencie.

Ponieważ u podstaw działania algorytmu genetycznego leżą losowo zachodzące zmiany (mutacje), a początkowe populacje agentów także generowane są w sposób losowy, poszczególne uruchomienia symulacji mogą się znacznie różnić. Celem zminimalizowania wpływu czynnika losowego na uzyskiwane wyniki, większość z przeprowadzonych badań zostało powtórzone 10-krotnie, a miary uzyskane z ich przebiegu uśredniono. Wyjątkiem są badania doboru parametrów i metod algorytmu genetycznego, które ze względu na swoją mnogość, zostały powtórzone 5-krotnie. Dzięki tym zabiegom, porównanie uzyskiwanych wyników jest bardziej wiarygodne.

Badania przeprowadzone w ramach niniejszej pracy zostały podzielone na serie. W każdej z nich wykonano szereg eksperymentów, których zadaniem jest sprawdzenie założeń postawionych w danej serii. Dodatkowo, każda seria (oprócz ostatniej) jest podzielona ze względu na eksperymenty dotyczące sterownika wykorzystującego sieć RNN oraz LSTM. Łącznie wszystkie badania zostaną przeprowadzone w ramach VI serii.

Seria I zawiera badania wstępne, których celem było znalezienie parametrów algorytmu genetycznego, dla których proces ewolucji zachodzi jak najsukuteczniej. Zawarte w niej eksperymenty podzielono według testowanych metod oraz rodzaju sterownika.

Kolejne serie zawierają badania właściwe, których celem jest sprawdzenie tezy postawionej w pracy. W ostatniej, VI serii przeprowadzono badania, które w sposób bezpośredni porównują obydwa rodzaje sterownika (RNN i LSTM).

## 5.1. SERIA I - STROJENIE PARAMETRÓW ALGORYTMU GENETYCZNEGO

Przed przystąpieniem do badań właściwych, zostały przeprowadzone badania w kierunku znalezienia metod oraz parametrów algorytmu genetycznego, dzięki którym proces ewolucji umożliwia jak najsukuteczniejsze uczenie się agentów. Ogólny sposób działania algorytmu genetycznego został przedstawiony w rozdziale 3.4. Metody, które zostały poddane tym badaniom dotyczą selekcji osobników (rozdział 3.4), krzyżowania oraz mutacji genów (rozdział 4.2.2). Ponadto, sprawdzone zostały również parametry generowania początkowej populacji oraz wpływu dodatkowej selekcji elitarnej. Ze względu na znaczne różnice w budowie wewnętrznej sieci RNN i LSTM, a zwłaszcza ich reprezentacji w formie genotypu oraz jego długości, dobór parametrów przeprowadzono osobno dla każdego rodzaju sterownika.

Badania parametrów przeprowadzono w uproszczonym środowisku. Agenty ofiar zostały wyłączone z procesu uczenia, aby zminimalizować efekt losowy. W ewolucji brali udział wyłącznie drapieżnicy, a wyniki były oceniane na podstawie skuteczności polowania, z wykorzystaniem miary przeżywalności ofiar. Oznacza to, że im mniejsza jest jej wartość, tym skuteczniejszy proces ewolucji, ponieważ drapieżnicy lepiej polują. W tym przypadku, wykorzystanie pozostałych miar nie jest konieczne do skutecznego porównania wyników. Parametry środowiska wykorzystanego na potrzeby eksperymentów są przedstawione w tabeli 4.4, natomiast zastosowane parametry algorytmu genetycznego zostały opisane przy każdym z testów oraz przedstawione w osobnych tabelach. W środowisku testowym obecni są dwaj drapieżnicy, dzięki czemu różnice między poszczególnymi wynikami powinny być bardziej widoczne.

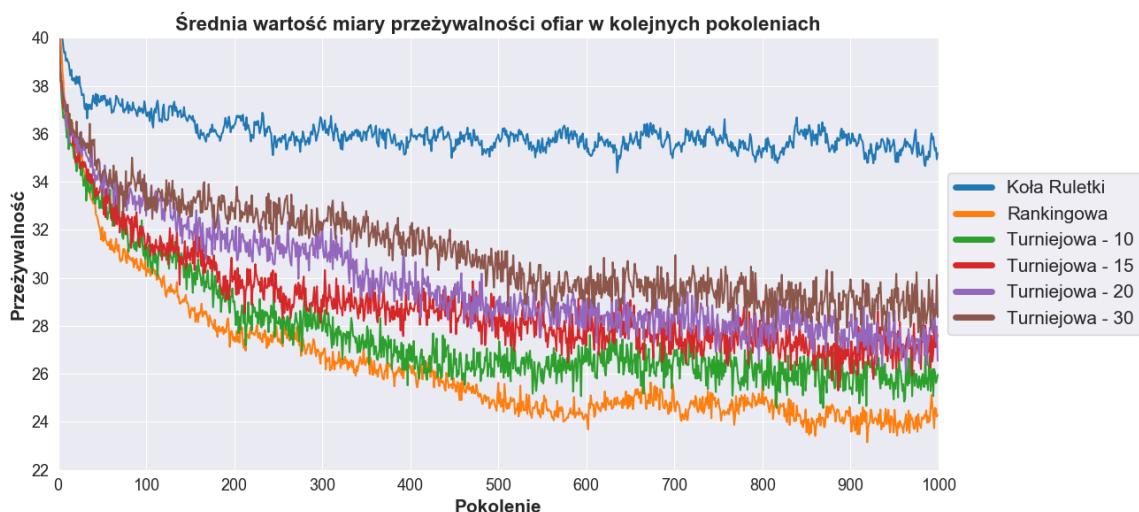
### 5.1.1. BADANIA WPŁYWU PARAMETRÓW ALGORYTMU GENETYCZNEGO DLA STEROWNIKA BĘDĄCEGO SIECIĄ RNN

#### Eksperyment 1 - Badanie wpływu metody selekcji osobników w algorytmie genetycznym na skuteczność ewolucji przy sterowniku wykorzystującym sieć RNN

Badania doboru parametrów rozpoczęto od metody selekcji (opisanych w rozdziale 3.4), ponieważ ma ona bardzo istotny wpływ na przebieg procesu uczenia się. Wartości pozostałych parametrów algorytmu genetycznego ustalone według tych zastosowanych w pracach [12, 15, 16] i przedstawiono w tabeli 5.1. W przypadku metody *turniejowej*, występuje dodatkowy parametr  $n$  odpowiedzialny za rozmiar pojedynczego turnieju, którego wartość także podlega badaniu, dlatego w przedstawieniu wyników zastosowano następujące oznaczenie: *Turniejowa - n*.

Tabela 5.1. Parametry algorytmu genetycznego dla eksperymentu 1 serii I

Nazwa parametru	Wartość
Rodzaj sterownika i liczba neuronów ukrytych	RNN - 8
Rozmiar populacji ofiar i drapieżników	100
Początkowe wartości genów	$\mathcal{N}(0, 1)$
Początkowe wypełnienie genotypu	100%
Metoda selekcji	parametr badany
Liczność selekcji elitarnej	0
Metoda krzyżowania	brak krzyżowania
Prawdopodobieństwo krzyżowania	0%
Prawdopodobieństwo mutacji pojedynczego genu	1%
Prawdopodobieństwo duplikacji genów	5%
Prawdopodobieństwo wypełnienia genów	0%
Prawdopodobieństwo usunięcia genów	2%



Rys. 5.1. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie metod selekcji dla sterownika RNN.

## Wyniki

Wyniki eksperymentu zostały zamieszczone na rysunku 5.1. Wykresy przedstawiają wartości średniej miary przeżywalności ofiar w kolejnych pokoleniach, uzyskanych dla różnych metod selekcji zastosowanych w algorytmie genetycznym. Dla selekcji *turniejowej* przeprowadzono badania dla rozmiaru turnieju wynoszącego [10, 15, 20, 30].

## Komentarz

Na podstawie wyników eksperymentu można zauważyc, że metoda koła ruletki wypadła najsłabiej, a jej wyniki znacznie odstają od pozostałych. W przypadku metody *turniejowej*, wyniki zauważalnie różnią się w zależności od zastosowanego rozmiaru turnieju (im mniejszy tym lepsze rezultaty uzyskano). Najlepsze wyniki otrzymano dla metody *rankingowej*, zastosowanej również w pracach [12], [15], [16], dlatego została wybrana do dalszych badań.

Metoda *koła ruletki* oraz *turniejowa*, zapewniają większą różnorodność genetyczną populacji, dzięki dopuszczeniu do kolejnego pokolenia również słabszych osobników, co zazwyczaj pomaga uniknąć problemu tak zwanego "lokalnego minima". Mimo to, wypadły w porównaniu znacznie słabiej od selekcji *rankingowej*, polegającej na wyborze połowy najlepszych osobników z danej populacji. Może to sugerować, że ewolucja wirtualnych agentów zachodzi podobnie jak w środowisku naturalnym, w której proces naturalnej selekcji faworyzuje właśnie najlepszych osobników, dlatego takie podejście okazało się najskuteczniejsze.

Tabela 5.2. Parametry algorytmu genetycznego dla eksperymentu 2 serii I

Nazwa parametru	Wartość
Rodzaj sterownika i liczba neuronów ukrytych	RNN - 8
Rozmiar populacji ofiar i drapieżników	100
Początkowe wartości genów	$\mathcal{N}(0, 1)$
Początkowe wypełnienie genotypu	100%
Metoda selekcji	<i>rankingowa</i>
Liczność selekcji elitarnej	0
Metoda krzyżowania	<i>brak krzyżowania</i>
Prawdopodobieństwo krzyżowania	0%
Prawdopodobieństwo mutacji pojedynczego genu	<i>parametr badany</i>
Prawdopodobieństwo duplikacji genów	<i>parametr badany</i>
Prawdopodobieństwo wypełnienia genów	<i>parametr badany</i>
Prawdopodobieństwo usunięcia genów	<i>parametr badany</i>

### Eksperyment 2 - Badanie wpływu różnych operatorów mutacji w algorytmie genetycznym na skuteczność ewolucji przy sterowniku wykorzystującym sieć RNN

Kolejnym etapem badań jest dobór metod mutacji oraz wartości ich prawdopodobieństw, które zostały dokładnie opisane w rozdziale 4.2.2. Wartości pozostałych parametrów przyjętych na potrzeby eksperymentu przedstawia tabela 5.2.

Niektóre z wyników zostały dodatkowo uśrednione dla wartości prawdopodobieństwa danej metody. Dla przykładu  $M1D[1 - 3]$  oznacza, że wyniki uzyskane z zastosowaniem metody  $M1$  uśredniono dla metody duplikacji  $D$  o wartościach [1%, 2%, 3%].

W celu uproszczenia zapisu, przy prezentacji wyników zastosowano następujące oznaczenia:

- $Mp$  - mutacja pojedynczego genu,
- $Dp$  - metoda duplikacji genów,
- $Dep$  - metoda usunięcia genów,
- $Fp$  - metoda wypełnienia genów,

gdzie  $p$  jest prawdopodobieństwem (w procentach) zajścia danej metody. Dla przykładu:  $M1De2$  oznacza, że zastosowano metodę mutacji pojedynczego genu o prawdopodobieństwie 1% oraz metodę usunięcia genów o prawdopodobieństwie 2%.

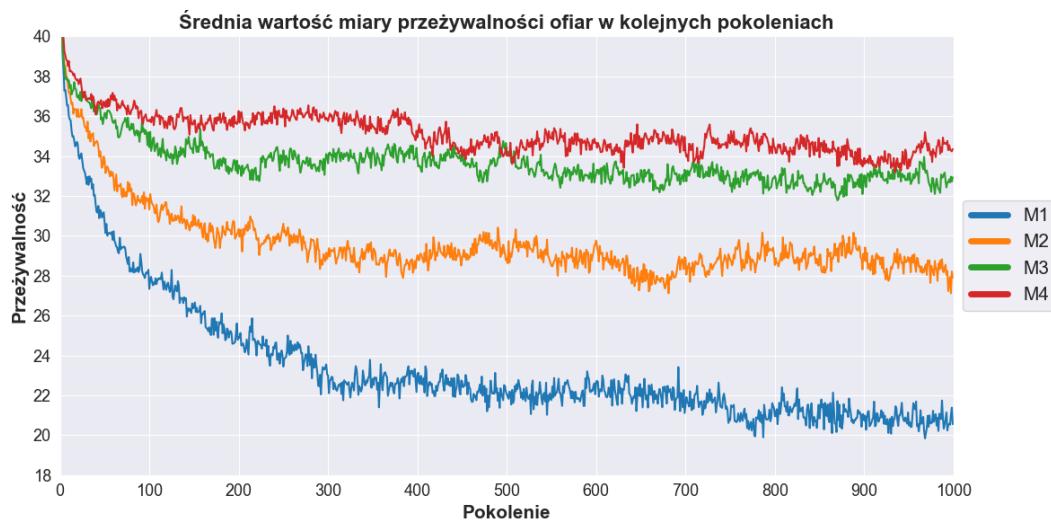
### Wyniki

Wyniki przeprowadzonego eksperymentu zostały zobrazowane na rysunkach 5.2-5.6, przedstawiających wykresy średniej przeżywalności ofiar. Możliwych kombinacji różnych metod oraz prawdopodobieństw mutacji jest bardzo wiele, dlatego zamieszczenie rezultatów uzyskanych dla wszystkich przeprowadzonych testów nie byłoby możliwe. W związku z tym, na wykresach przedstawiono wyłącznie najważniejsze wyniki.

Rysunek 5.2 przedstawia rezultaty uzyskane dla metody mutacji  $M$  o wartości prawdopodobieństwa [1%, 2%, 3%, 4%]. Na rysunku 5.3 zamieszczono rezultaty uzyskane przy równoczesnym zastosowaniu  $M1$  oraz każdej z pozostałych metod ( $D$ ,  $F$ ,  $De$ ). Wyniki zostały uśrednione dla prawdopodobieństwa drugiej z metod wynoszącego od 1% do 10% z krokiem 1%.

Rysunek 5.4 przedstawia wyniki wybranych kombinacji, uśrednionych dla prawdopodobieństw metod  $De$  oraz  $F$ , wynoszących [1%, 2%, 3%, 4%]. Na rysunku 5.5 zobrazowano te same dane, jednak uśrednione względem różnych wartości prawdopodobieństwa metody  $D$ .

Najlepsze z uzyskanych wyników dla przetestowanych kombinacji metod mutacji genotypu zamieszczono na rysunku 5.6.



Rys. 5.2. Wykres średniej wartości przeżywalności ofiar w kolejnych generacjach. Porównanie prawdopodobieństwa pojedynczej mutacji genu dla sterownika RNN.

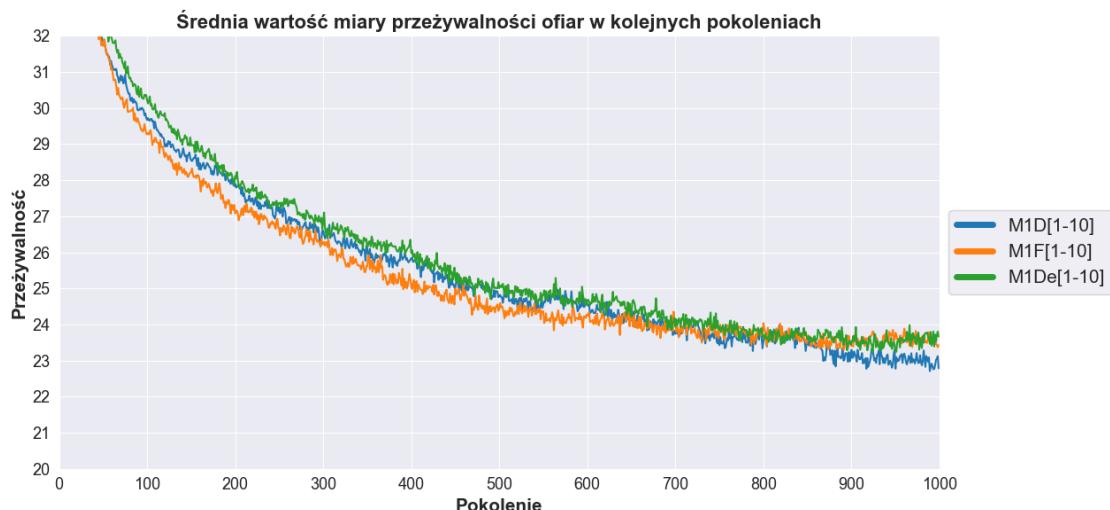
## Komentarz

Według uzyskanych wyników, największy wpływ na skuteczność uczenia się ma metoda pojedynczej mutacji genu. Zdecydowanie najlepiej radzi sobie  $M1$ , natomiast wraz ze wzrostem prawdopodobieństwa jej wystąpienia, obserwuje się znaczny spadek jakości otrzymywanych rezultatów. Z tego względu wybrano ją do dalszych badań.

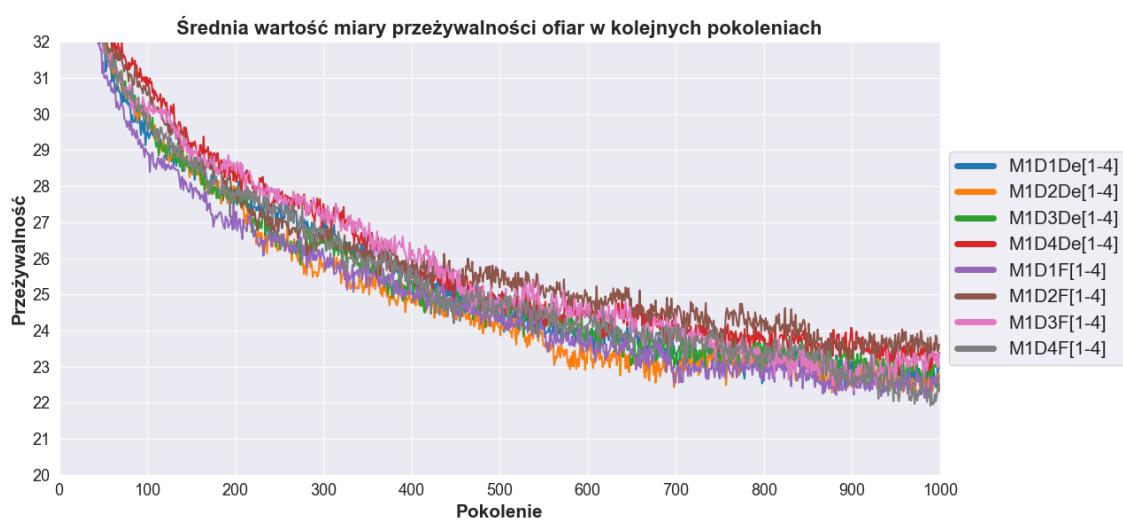
Zastosowanie drugiej metody mutacji (dodatkowo do  $M1$ ), nie przyniosło znacznej poprawy uzyskiwanych wyników. Każda z dodatkowych metod ( $D$ ,  $De$ ,  $F$ ), w średnim przypadku daje dość zbliżone rezultaty (Rys. 5.3). Można zauważyć, że  $F$  powodowała nieco szybszą ewolucję na początku, jednak z czasem ta różnica maleje i sytuacja zmienia się na korzyść  $D$ , która w przeciwieństwie do pozostałych metod mutacji, operuje na już istniejących genach (kopując je w nowe miejsce). Ze względu na to, zdecydowano się na wybór  $D$  jako drugiej metody mutacji.

Następnie przebadano trzy różne metody mutacji jednocześnie. Niezależnie od prawdopodobieństwa metody  $D$ , uzyskiwane wyniki były bardzo zbliżone (Rys. 5.5). Te same wyniki (Rys. 5.5), lecz uśrednione względem wartości prawdopodobieństwa  $D$  pokazały, że dla  $De2$  oraz  $F3$ , uzyskiwane wartości są nieco lepsze od pozostałych. Przeprowadzono również badania przy uwzględnieniu wszystkich czterech metod mutacji, których wyników nie zamieszczono ze względu na ich dużą liczbę.

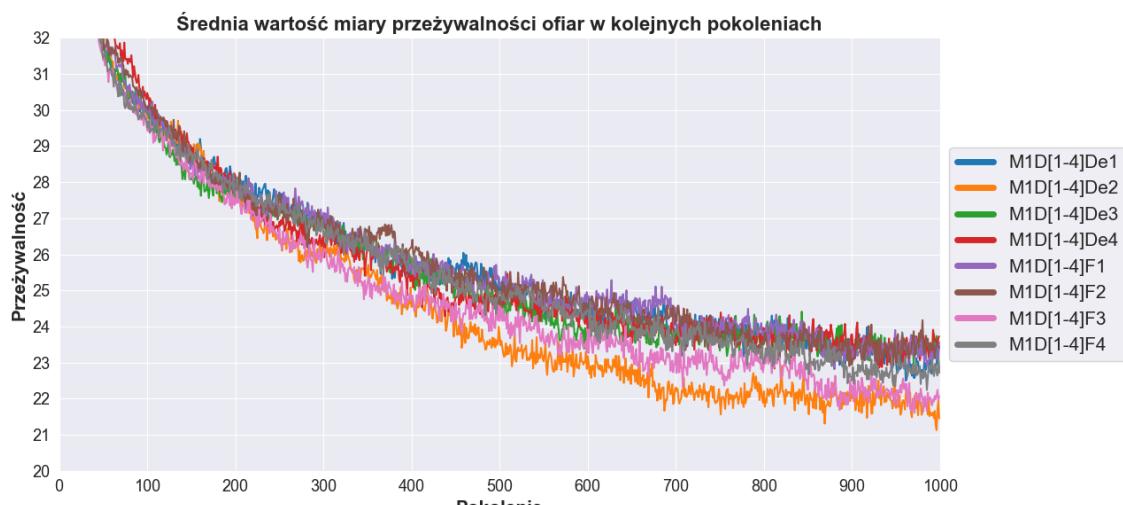
Przy zestawieniu najlepszych ze wszystkich przebadanych mutacji (Rys. 5.6), można zauważyć, że wyróżnia się metoda mutacji pojedynczego genu  $M1$ , której wyniki na początkowym etapie ewolucji są znacznie lepsze od pozostałych kombinacji, jednak z czasem ta przewaga zatracza się, a ostatecznie uzyskiwane rezultaty są do siebie dość zbliżone. Może to wynikać z tego, że  $M$  działa na pojedyncze geny, dzięki czemu zachodzące mutacje są bardziej ostrożne w przeciwieństwie do pozostałych metod, które operują na fragmentach genotypu, co zwiększa szansę na niekorzystną mutację i w konsekwencji może spowolnić proces ewolucji. W związku z tym, wybrane kombinacje zostały poddane dodatkowym badaniom wraz z metodą krzyżowania w eksperymencie 3.



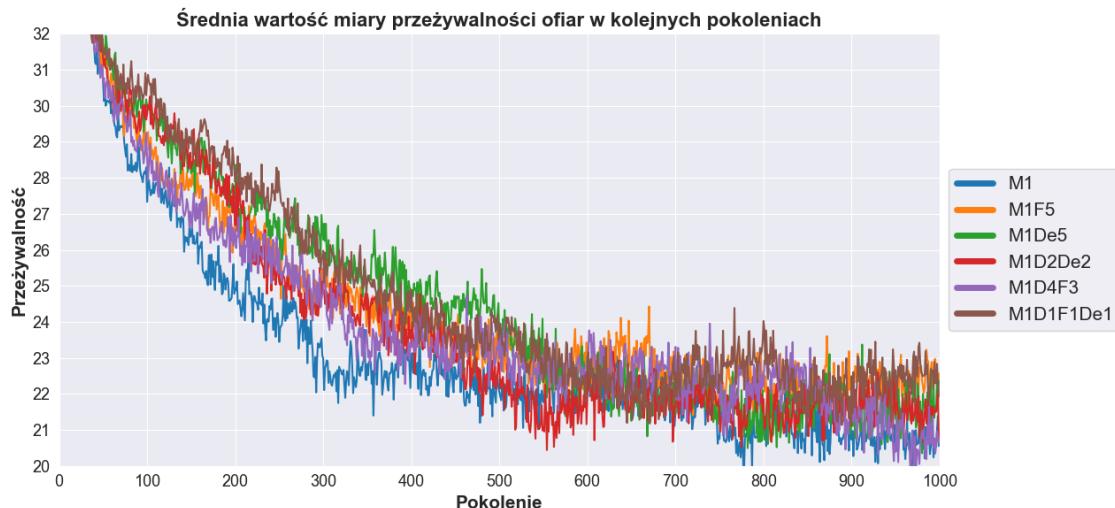
Rys. 5.3. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie metod mutacji dla sterownika RNN.



Rys. 5.4. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie metod mutacji dla sterownika RNN.



Rys. 5.5. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie metod mutacji dla sterownika RNN.



Rys. 5.6. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie najlepszych z przebadanych kombinacji mutacji dla sterownika RNN.

Tabela 5.3. Parametry algorytmu genetycznego dla eksperymentu 3 serii I

Nazwa parametru	Wartość
Rodzaj sterownika i liczba neuronów ukrytych	RNN - 8
Rozmiar populacji ofiar i drapieżników	100
Początkowe wartości genów	$\mathcal{N}(0, 1)$
Początkowe wypełnienie genotypu	100%
Metoda selekcji	rankingowa
Liczność selekcji elitarnej	0
Metoda krzyżowania	parametr badany
Prawdopodobieństwo krzyżowania	parametr badany
Prawdopodobieństwo mutacji pojedynczego genu	parametr badany
Prawdopodobieństwo duplikacji genów	parametr badany
Prawdopodobieństwo wypełnienia genów	parametr badany
Prawdopodobieństwo usunięcia genów	parametr badany

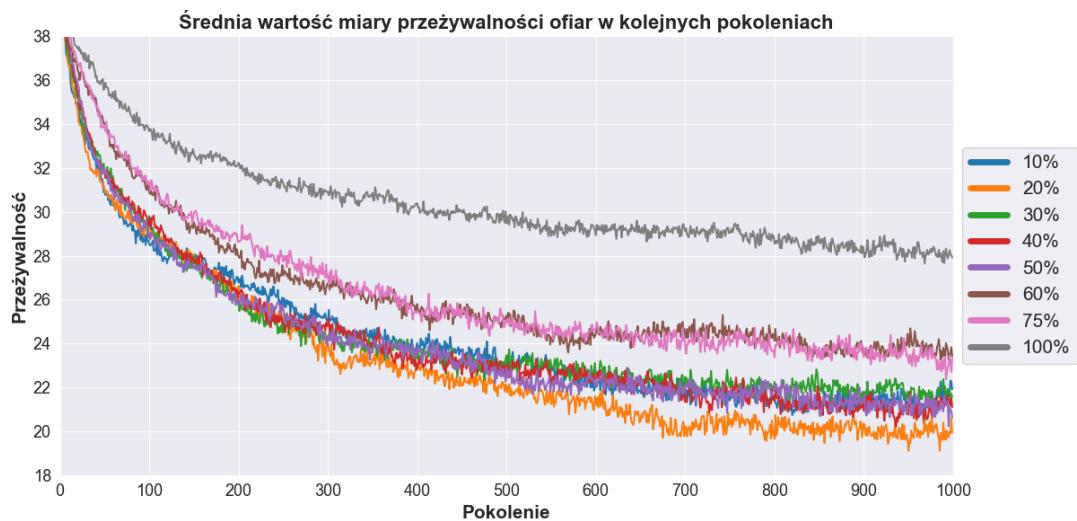
### Eksperyment 3 - Badanie wpływu metody krzyżowania w algorytmie genetycznym na skuteczność ewolucji przy sterowniku wykorzystującym sieć RNN

W pracach [12], [15], [16], algorytm genetyczny opierał swoje działanie wyłącznie na operatorach mutacji genów. W praktyce często stosowana jest również jedna z metod krzyżowania (przedstawionych w rozdziale 4.2.2). W tym eksperymencie przeprowadzone zostaną testy metod oraz prawdopodobieństwa zajścia krzyżowania, dla najlepszych kombinacji mutacji uzyskanych w eksperymencie 2. Pozostałe parametry przedstawia tabela 5.3.

W celu uproszczenia zapisu, przy prezentacji wyników zastosowano następujące oznaczenia:

- $SPp$  - krzyżowanie jedno-punktowe (ang. *Single-Point*),
- $TPp$  - krzyżowanie dwu-punktowe (ang. *Two-Point*),
- $SPPp$  - krzyżowanie jedno-punktowe na każdy segment (ang. *Single-Point per Part*),
- $TPPp$  - krzyżowanie dwu-punktowe na każdy segment (ang. *Two-Point per Part*),

gdzie  $p$  jest prawdopodobieństwem (w procentach) zajścia danego krzyżowania.



Rys. 5.7. Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie wartości prawdopodobieństwa krzyżowania dla sterownika RNN.

## Wyniki

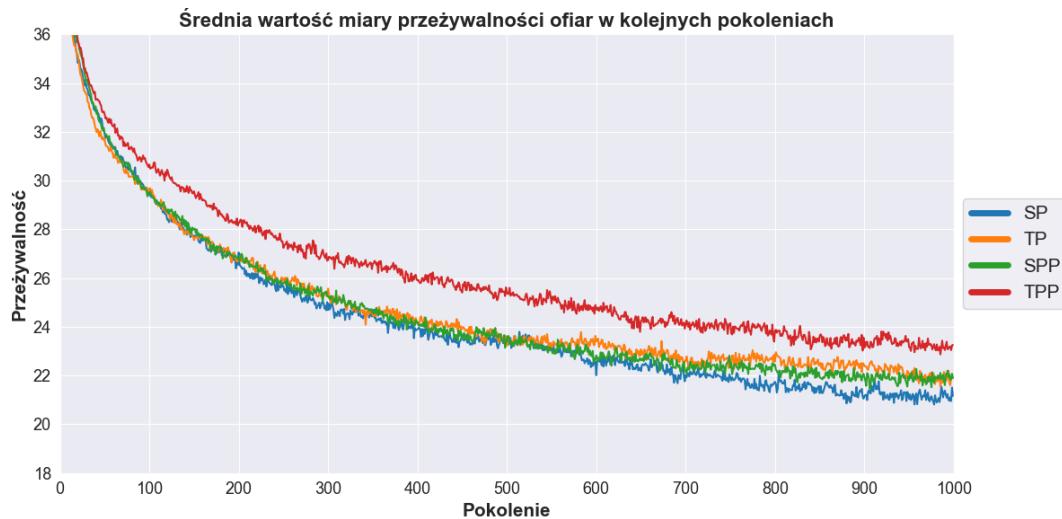
Wyniki eksperymentu przedstawiono na rysunkach 5.7-5.9 w formie wykresów średniej miary przeżywalności ofiar. Jako pierwsza badaniom poddana została metoda mutacji *M1* wraz wymienionymi metodami krzyżowania dla wartości prawdopodobieństwa [10%-60% z krokiem co 10%, 75%, 100%]. Uzyskane rezultaty zostały uśrednione dla każdego z prawdopodobieństw (Rys. 5.7). Następnie badaniom poddano pozostałe z wybranych mutacji (*M1D1F1*, *M1D2De2*, *M1D1F1De1*) wraz z krzyżowaniem dla prawdopodobieństw [20%, 30%, 40%]. Wyniki zostały uśrednione dla poszczególnych metod krzyżowania (Rys. 5.8).

Najlepsze z uzyskanych wyników dla każdej z metod krzyżowania zamieszczono na rysunku 5.9. Dla porównania przedstawiono również wyniki bez krzyżowania.

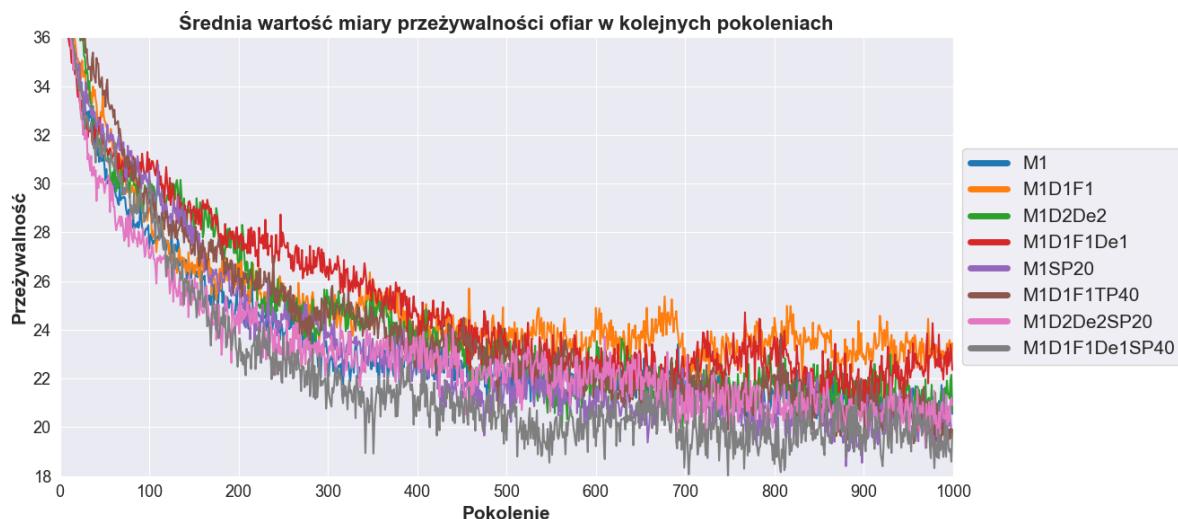
## Komentarz

Według obserwacji, przy zbyt wysokich prawdopodobieństwach krzyżowania, uzyskiwane rezultaty są znacznie słabsze, a w niektórych przypadkach (nieuśrednionych), całkowicie uniemożliwiały postęp w procesie ewolucji. Krzyżowanie jest procesem, który w dużym stopniu oddziałuje na genotyp, co może prowadzić do niekorzystnych zmian, a przy zbyt częstych krzyżowaniach, znacznie spowolnić proces ewolucji. Może to być przyczyną powyższej obserwacji. Najlepiej w porównaniu wypada prawdopodobieństwo równe 20%, jednak inne bliskie wartości (10%, 30%, 40%) nie są o wiele gorsze. W związku z tym, pozostałe kombinacje mutacji przebadano tylko dla tych wartości.

Można zauważyć, że z zastosowanych metod krzyżowania, najsłabiej sprawuje się *TppP*. Wyniki uzyskiwane dla pozostałych są do siebie zbliżone, jednak nieco lepiej wypadła metoda *SP*. W ogólnym przypadku, krzyżowanie prowadziło do poprawy rezultatów tylko dla niektórych kombinacji. Najlepszy wyniki uzyskano dla *M1D1F1De1* wraz z krzyżowaniem *SP30*, w związku z czym zostaje ona wybrana do dalszych badań.



Rys. 5.8. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie metod krzyżowania dla sterownika RNN.



Rys. 5.9. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie najlepszych z przebadanych kombinacji dla sterownika RNN.

Tabela 5.4. Parametry algorytmu genetycznego dla eksperymentu 4 serii I

Nazwa parametru	Wartość
Rodzaj sterownika i liczba neuronów ukrytych	RNN - 8
Rozmiar populacji ofiar i drapieżników	100
Początkowe wartości genów	parametr badany
Początkowe wypełnienie genotypu	parametr badany
Metoda selekcji	rankingowa
Liczność selekcji elitarnej	0
Metoda krzyżowania	jedno-punktowa
Prawdopodobieństwo krzyżowania	30%
Prawdopodobieństwo mutacji pojedynczego genu	1%
Prawdopodobieństwo duplikacji genów	1%
Prawdopodobieństwo wypełnienia genów	1%
Prawdopodobieństwo usunięcia genów	1%

## Eksperyment 4 - Badanie wpływu inicjalizacji wartości genotypu w algorytmie genetycznym na skuteczność ewolucji przy sterowniku wykorzystującym sieć RNN

Podczas generacji nowego pokolenia, początkowe wartości genotypów są ustalane losowo. Pojawienie się korzystnych genów już na tym etapie może znacznie przyspieszyć proces ewolucji, dlatego sposób ich inicjalizacji jest ważny. Do tej pory, w tym celu wykorzystywano wartości pochodzące z rozkładu normalnego  $\mathcal{N}(0, 1)$ , a genotyp był nimi wypełniany w 100%. W tym eksperymencie przebadane zostaną wartości parametru odchylenia standardowego  $\sigma^2$  dla rozkładu normalnego oraz stopień początkowego wypełnienia genotypu  $St_{init}$  wartościami różnymi od zera (pochodzącymi z rozkładu  $\mathcal{N}(0, \sigma^2)$ ). Pozostałe parametry przedstawia tabela 5.4

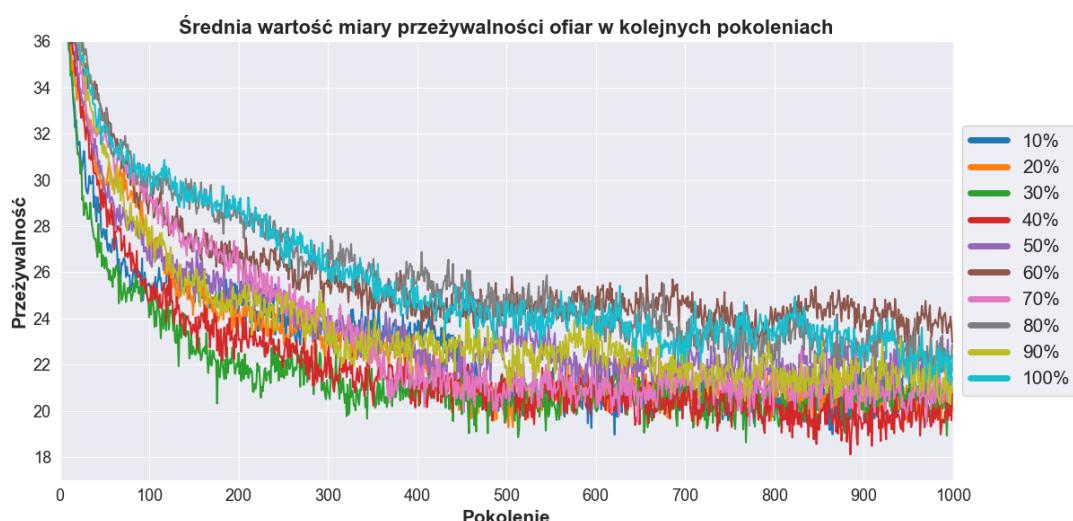
### Wyniki

Wyniki uzyskane w ramach eksperymentu, przedstawiające miarę średniej przeżywalności ofiar zamieszczono na rysunkach 5.10 i 5.11. W pierwszej kolejności badaniom poddano stopień początkowego wypełnienia genotypu dla wartości 10% - 100% z krokiem co 10% (Rys. 5.10). Następnie wybrano  $St_{init} = 30\%$  oraz przeprowadzono badania parametru  $\sigma^2$  dla wartości wynoszących [0.5, 1, 2, 3, 5, 7] (Rys. 5.11).

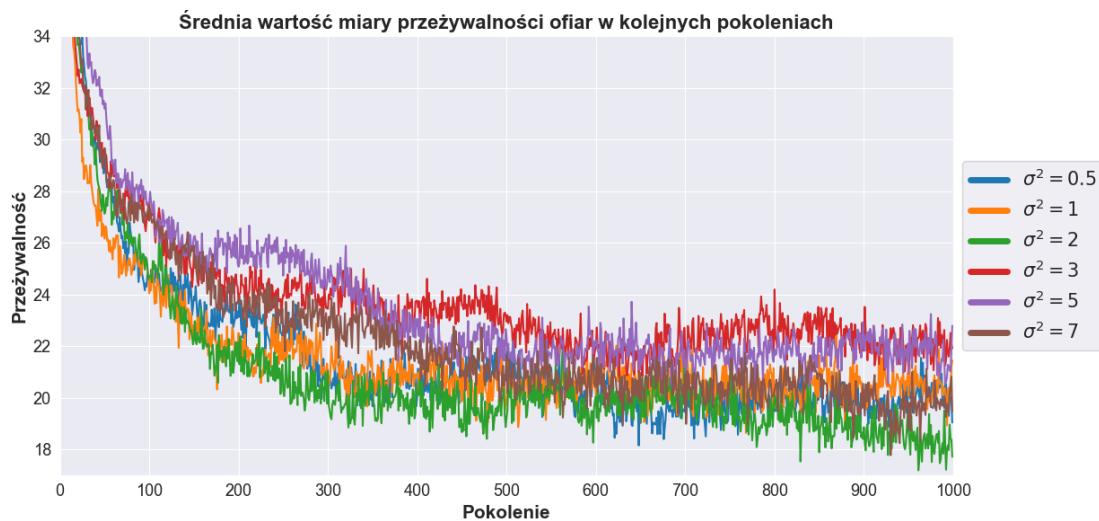
### Komentarz

Na podstawie uzyskanych wyników, można zauważać większą skuteczność procesu ewolucji dla mniejszych wartości parametru  $St_{init}$  (10%-40%). Różnica widoczna jest zwłaszcza w początkowych pokoleniach, gdzie uczenie przebiega zauważalnie skuteczniej i szybciej prowadzi do osiągnięcia lepszych wyników. Może to wynikać z tego, że prostszym jest utworzenie nowych i pozytywnych genów, bazując na genotypie wypełnionym w niewielkim stopniu, niż eliminacja już istniejących, lecz negatywnych cech, częściej obecnych w losowo wypełnionym genotypie. Ze względu na to, do dalszych badań została wybrana wartość równa 30%, która najbardziej podniosła skuteczność ewolucji.

W przypadku parametru  $\sigma^2$ , uzyskane rezultaty są dość zbliżone względem wartości  $\sigma^2 = 1$  (wykorzystywanej w poprzednich eksperymentach). Dla wartości (3, 5, 7) można zauważać niewielkie pogorszenie wyników, lecz może to wynikać z przypadku. Dla  $\sigma^2 = 2$  zaobserwowano poprawę skuteczności uczenia w końcowych pokoleniach, dlatego ostatecznie zdecydowano się na wybór rozkładu  $\mathcal{N}(0, 2)$ .



Rys. 5.10. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie wartości początkowego wypełnienia genotypu  $St_{init}$  dla sterownika RNN.



Rys. 5.11. Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie parametru  $\sigma^2$  z rozkładu normalnego dla sterownika RNN.

Tabela 5.5. Parametry algorytmu genetycznego dla eksperymentu 5 serii I

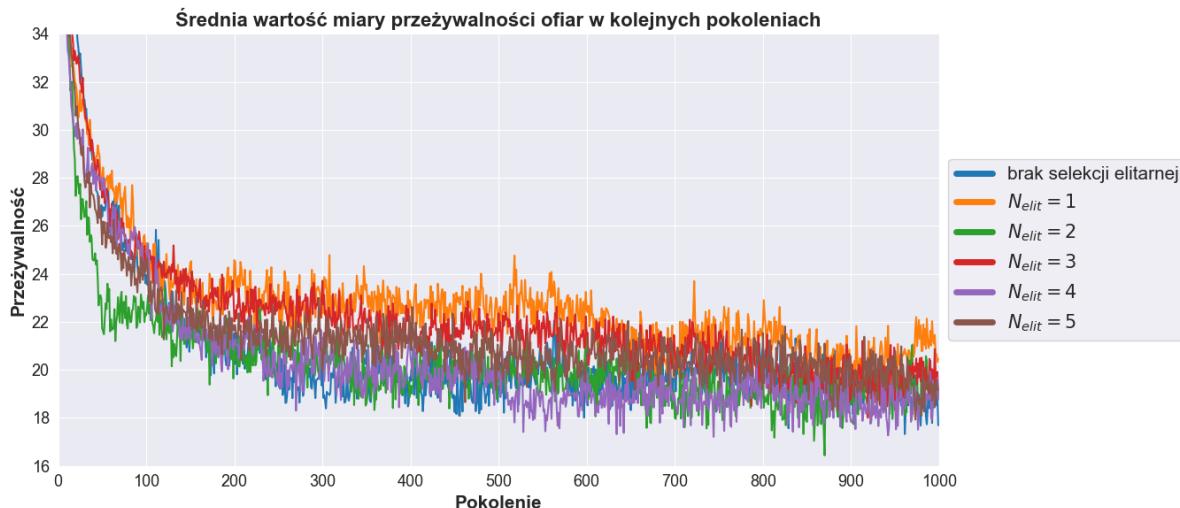
Nazwa parametru	Wartość
Rodzaj sterownika i liczba neuronów ukrytych	RNN - 8
Rozmiar populacji ofiar i drapieżników	100
Początkowe wartości genów	$\mathcal{N}(0, 2)$
Początkowe wypełnienie genotypu	30%
Metoda selekcji	<i>rankingowa</i>
Liczność selekcji elitarnej	<i>parametr badany</i>
Metoda krzyżowania	<i>jedno-punktowa</i>
Prawdopodobieństwo krzyżowania	30%
Prawdopodobieństwo mutacji pojedynczego genu	1%
Prawdopodobieństwo duplikacji genów	1%
Prawdopodobieństwo wypełnienia genów	1%
Prawdopodobieństwo usunięcia genów	1%

### Eksperyment 5 - Badanie wpływu metody selekcji elitarnej w algorytmie genetycznym na skuteczność ewolucji przy sterowniku wykorzystującym sieć RNN

W niektórych przypadkach, w algorytmie genetycznym stosowana jest również selekcja elitarna, opisana w rozdziale 3.4. Występuje w niej dodatkowy parametr  $N_{elit}$ , określający liczbę osobników, którzy w niezmiennej formie trafią do kolejnej populacji. W tym eksperymencie zostały przetestowane różne wartości parametru  $N_{elit}$ . Wartości parametrów algorytmu genetycznego, ustalone według poprzednich eksperymentów przedstawia tabela 5.5.

#### Wyniki

Wyniki przedstawiające miarę średniej przeżywalności ofiar na przestrzeni kolejnych pokoleń, w zależności od wartości  $N_{elit}$ , zobrazowano na rysunku 5.12. Parametr  $N_{elit}$  został przebadany dla wartości od 1 do 5 z krokiem co 1. Dla porównania zamieszczono również rezultaty uzyskane bez wykorzystania metody selekcji elitarnej.



Rys. 5.12. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie rozmiaru selekcji elitarnej  $N_{elit}$  dla sterownika RNN.

### Komentarz

Niezależnie od wartości  $N_{elit}$ , skuteczność ewolucji nie uległa znaczącej poprawie, a w niektórych przypadkach została nawet pogorszona. Może to wynikać z zastosowanej już selekcji *rankingowej*, która i tak wybiera połowę najlepszych osobników z populacji, dlatego dodatkowa selekcja elitarna nie poprawia wyników. Jednak dla  $N_{elit} = 2$  uczenie na początkowym etapie zachodziło nieco szybciej, dlatego zdecydowano się na wprowadzenie do algorytmu genetycznego metody selekcji elitarnej o wartości 2.

### 5.1.2. BADANIA WPŁYWU PARAMETRÓW ALGORYTMU GENETYCZNEGO DLA STEROWNIKA BĘDĄCEGO SIECIĄ LSTM

#### Eksperyment 6 - Badanie wpływu metody selekcji osobników w algorytmie genetycznym na skuteczność ewolucji przy sterowniku wykorzystującym sieć LSTM

Analogicznie jak w przypadku sterownika RNN, badania doboru parametrów rozpoczęto od metody selekcji osobników, opisanych w rozdziale 3.4. Wartości pozostałych parametrów algorytmu genetycznego ustalono według prac [12, 15, 16] i przedstawiono w tabeli 5.6. Metoda *turniejowa* posiada dodatkowy parametr  $n$  odpowiedzialny za rozmiar pojedynczego turnieju, którego wartość także podlega badaniu, dlatego w przedstawieniu wyników zastosowano następujące oznaczenie: *Turniejowa - n*.

### Wyniki

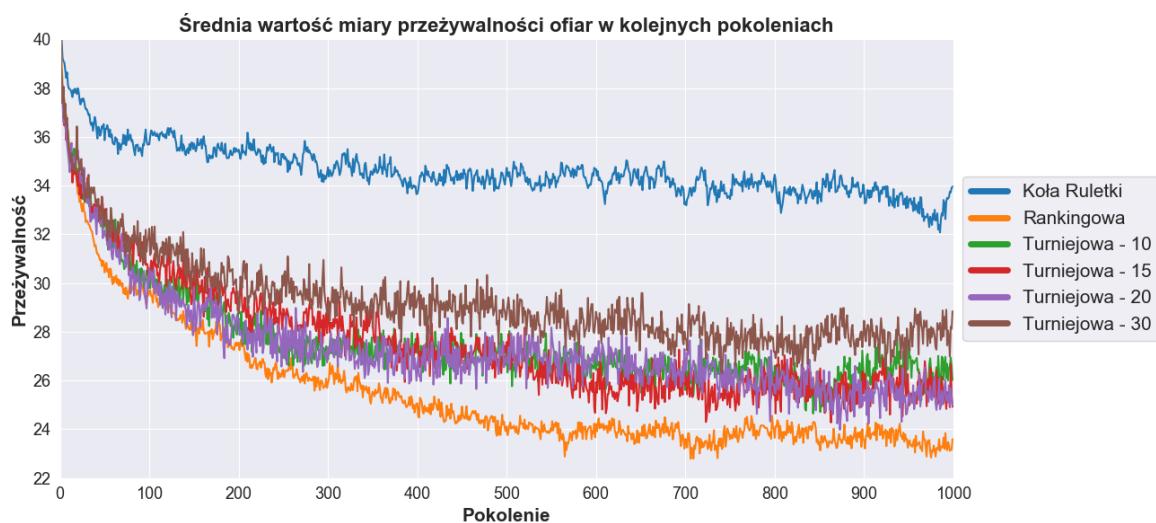
Wyniki eksperymentu przedstawia rysunek 5.13. Zawarte na nim wykresy obrazują wartości średniej miary przeżywalności ofiar w kolejnych pokoleniach dla różnych metod selekcji populacji w algorytmie genetycznym. Dla selekcji *turniejowej* przeprowadzono badania dla rozmiaru turnieju wynoszącego [10, 15, 20, 30].

### Komentarz

Podobnie jak w przypadku sterownika RNN, metoda *koła ruletki* wypadła najsłabiej w porównaniu do pozostałych. Wyniki uzyskane dla metody *turniejowej* nieco różnią się w zależności od zastosowanego rozmiaru turnieju, jednak są one bardziej zbliżone do siebie niż w przypadku sieci RNN. Nie zaobserwowano również poprawy skuteczności uczenia się wraz ze zmniejszonym rozmiarem turnieju. Najlepsze wyniki ponownie uzyskano przy zastosowaniu metody *rankingowej*, dlatego została wybrana do dalszych badań.

Tabela 5.6. Parametry algorytmu genetycznego dla eksperymentu 6 serii I

Nazwa parametru	Wartość
Rodzaj sterownika i liczba neuronów ukrytych	LSTM - 8
Rozmiar populacji ofiar i drapieżników	100
Początkowe wartości genów	$\mathcal{N}(0, 1)$
Początkowe wypełnienie genotypu	100%
Metoda selekcji	parametr badany
Liczność selekcji elitarnej	0
Metoda krzyżowania	brak krzyżowania
Prawdopodobieństwo krzyżowania	0%
Prawdopodobieństwo mutacji pojedynczego genu	1%
Prawdopodobieństwo duplikacji genów	5%
Prawdopodobieństwo wypełnienia genów	0%
Prawdopodobieństwo usunięcia genów	2%



Rys. 5.13. Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod selekcji dla sterownika LSTM.

Tabela 5.7. Parametry algorytmu genetycznego dla eksperymentu 7 serii I

Nazwa parametru	Wartość
Rodzaj sterownika i liczba neuronów ukrytych	LSTM - 8
Rozmiar populacji ofiar i drapieżników	100
Początkowe wartości genów	$\mathcal{N}(0, 1)$
Początkowe wypełnienie genotypu	100%
Metoda selekcji	rankingowa
Liczność selekcji elitarnej	0
Metoda krzyżowania	brak krzyżowania
Prawdopodobieństwo mutacji pojedynczego genu	parametr badany
Prawdopodobieństwo duplikacji genów	parametr badany
Prawdopodobieństwo wypełnienia genów	parametr badany
Prawdopodobieństwo usunięcia genów	parametr badany

## Eksperyment 7 - Badanie wpływu różnych operatorów mutacji w algorytmie genetycznym na skuteczność ewolucji przy sterowniku wykorzystującym sieć LSTM

W tym eksperymencie przetestowano różne podejścia do mutacji genotypu, które zostały dokładnie opisane w rozdziale 4.2.2. Wartości pozostały parametrów algorytmu genetycznego przyjętych na potrzeby eksperymentu przedstawia tabela 5.7.

Niektóre z wyników zostały dodatkowo uśrednione dla wartości prawdopodobieństwa danej metody. Dla przykładu  $M1D[1 - 3]$  oznacza, że wyniki uzyskane z zastosowaniem metody  $M1$  uśredniono dla metody duplikacji  $D$  o wartościach [1%, 2%, 3%].

W celu uproszczenia zapisu, zastosowano następujące oznaczenia:

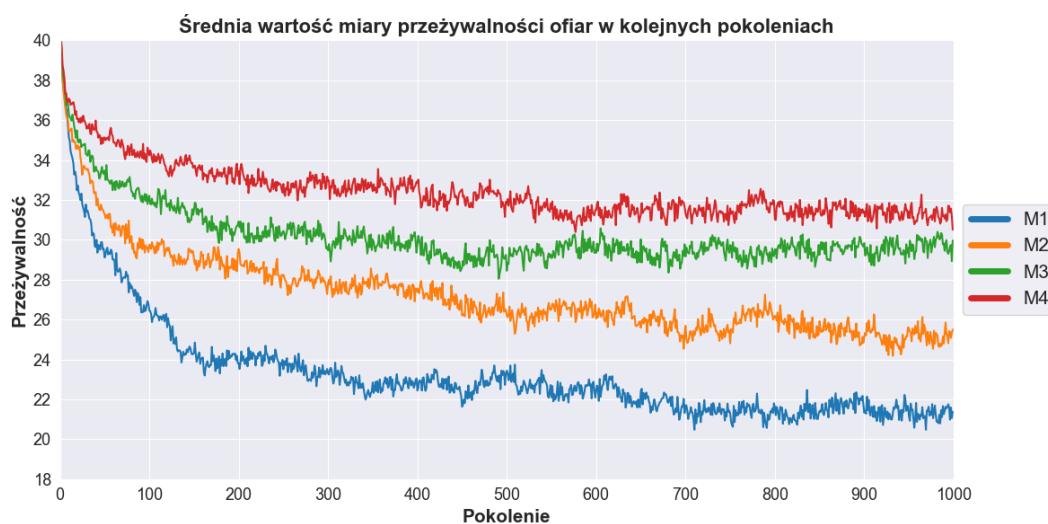
- $Mp$  - mutacja pojedynczego genu,
- $Dp$  - metoda duplikacji genów,
- $Dep$  - metoda usunięcia genów,
- $Fp$  - metoda wypełnienia genów,

gdzie  $p$  jest prawdopodobieństwem (w procentach) zajścia danej metody. Dla przykładu:  $M1De2$  oznacza, że zastosowano metodę mutacji pojedynczego genu o prawdopodobieństwie 1% oraz metodę usunięcia genów o prawdopodobieństwie 2%.

### Wyniki

Wyniki uzyskane w ramach eksperymentu przedstawiono na rysunkach 5.14-5.18 w formie wykresów miary średniej przeżywalności ofiar. Zdecydowano się na zamieszczenie wyłącznie najważniejszych wyników, ponieważ wszystkich kombinacji metod i prawdopodobieństwa mutacji jest zbyt wiele.

Rysunek 5.14 przedstawia wyniki uzyskane przy zastosowaniu pojedynczej mutacji genu  $M$  dla prawdopodobieństwa [1%, 2%, 3%, 4%]. Na rysunku 5.15 zamieszczono wyniki uzyskane z wykorzystaniem dwóch metod mutacji,  $M1$  oraz dodatkowo jednej z pozostałych ( $D$ ,  $F$ ,  $De$ ), które zostały uśrednione dla prawdopodobieństwa drugiej z metod wynoszącego od 1% do 10% z krokiem 1%.



Rys. 5.14. Wykres średniej wartości przeżywalności ofiar w kolejnych generacjach. Porównanie prawdopodobieństwa pojedynczej mutacji genu dla sterownika LSTM.

Na rysunku 5.16 przedstawiono wyniki wybranych metod mutacji, które uśredniono dla prawdopodobieństwa metody  $D$  oraz  $F$ , równego [1%, 2%, 3%, 4%]. Rysunek 5.17 obrazuje te same dane, jednak uśrednione względem różnych wartości prawdopodobieństwa metody  $D$ .

Wyniki kombinacji, dla których proces ewolucji przebiegał najskuteczniej zamieszczono na rysunku 5.18.

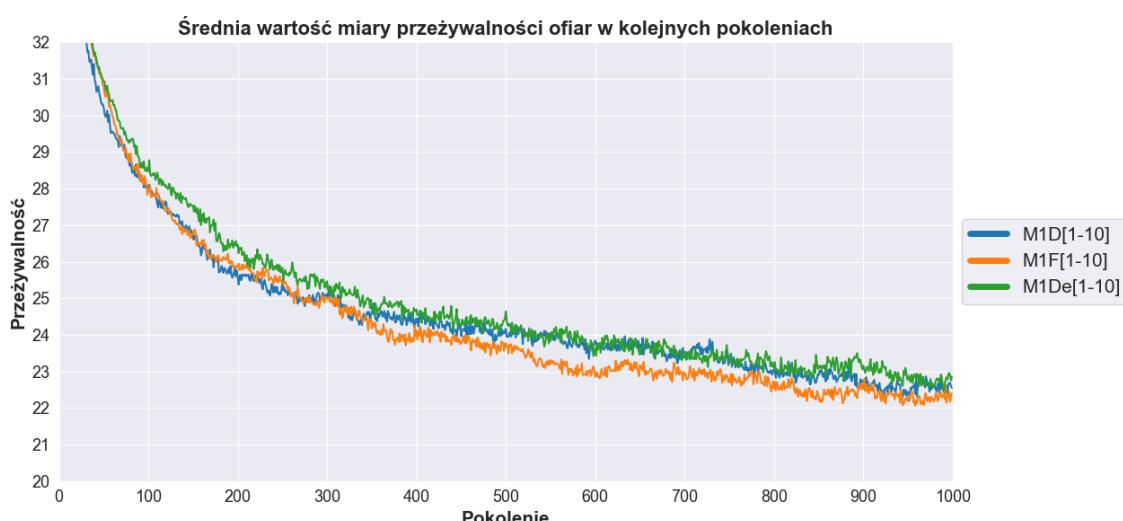
### Komentarz

Podobnie jak w przypadku sterownika RNN, największy wpływ na skuteczność uczenia się ma metoda pojedynczej mutacji genu z prawdopodobieństwem 1%, natomiast wraz ze wzrostem częstości jej wystąpienia zachodzi spadek jakości procesu ewolucji. Ze względu na to  $M1$  została wybrana do dalszych badań.

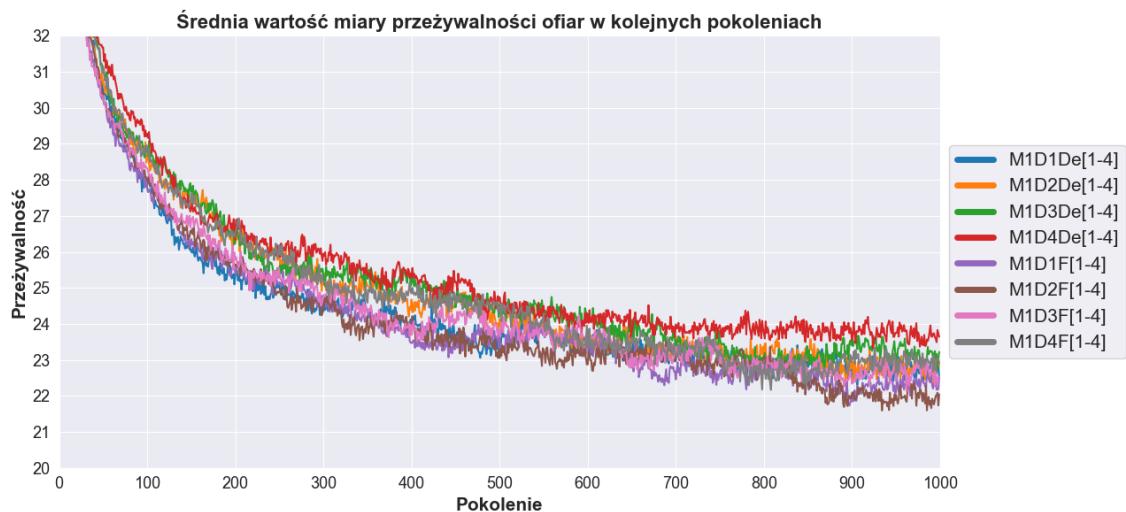
Wyniki uzyskane przy zastosowaniu dwóch różnych metod mutacji (Rys. 5.15), są do siebie zbliżone.  $F$  okazała się nieco skuteczniejsza w początkowych pokoleniach, jednak ta różnica maleje w końcowym etapie ewolucji. Tak jak dla sieci RNN, zdecydowano się na wybór  $D$  przez wzgląd na inny charakter tej metody (wykorzystuje już istniejące geny).

Po wprowadzeniu trzeciej metody mutacji, niezależnie od prawdopodobieństwa metody  $D$ , uzyskiwane rezultaty są zbliżone (Rys. 5.16), jednak nieco lepsze z wykorzystaniem metody  $F$ . Rysunek 5.17 przedstawia te same wyniki, lecz uśrednione względem prawdopodobieństwa metody  $D$ . W tym przypadku otrzymane rezultaty są do siebie jeszcze bardziej podobne, jednak wciążauważalna jest przewaga  $F$ . Przeprowadzone zostały również badania przy uwzględnieniu wszystkich czterech metod mutacji, jednak ich wyników nie zamieszczono, ze względu na ich dużą liczbę.

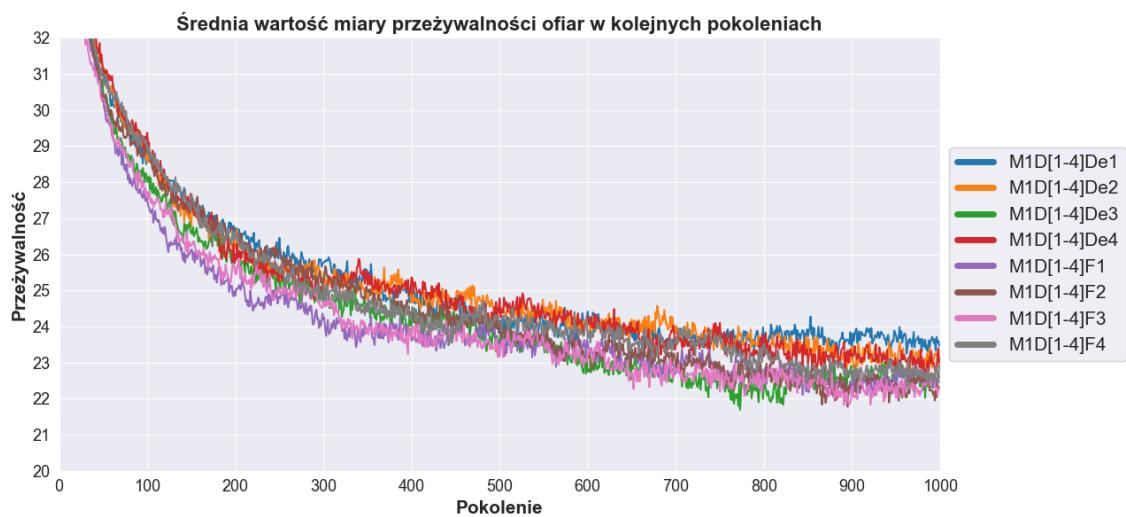
Na zestawieniu wyników uzyskanych dla najlepszych kombinacji mutacji (Rys. 5.18), można zauważać, że rezultaty są do siebie mocno zbliżone (nawet bardziej niż dla sterownika RNN), dlatego nie można jednoznacznie stwierdzić, które połączenie mutacji jest najskuteczniejsze. W związku z tym, wybrane kombinacje zostały poddane dodatkowym badaniom wraz z metodą krzyżowania w eksperymencie 8.



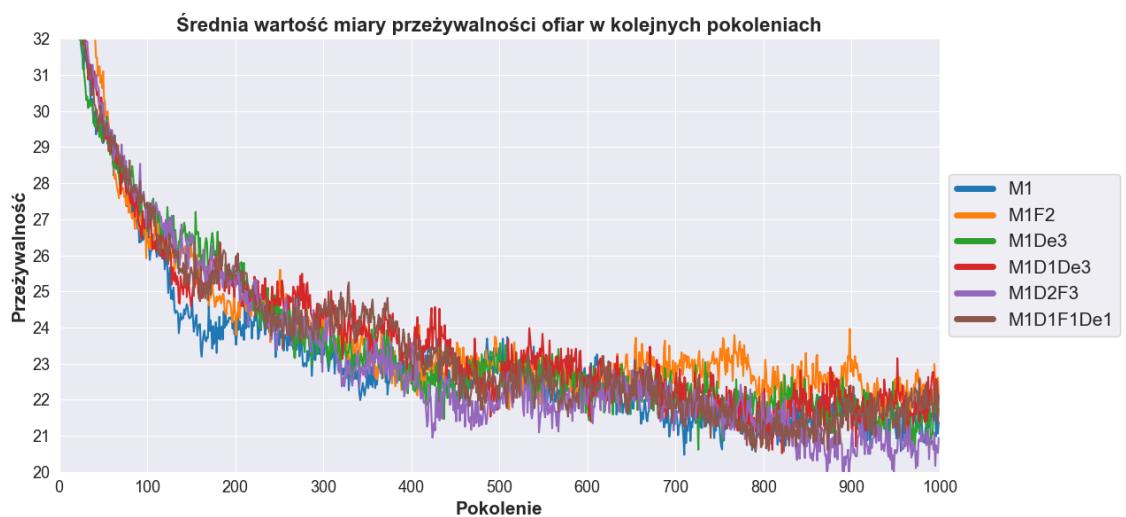
Rys. 5.15. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie metod mutacji dla sterownika LSTM.



Rys. 5.16. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie metod mutacji dla sterownika LSTM.



Rys. 5.17. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie metod mutacji dla sterownika LSTM.



Rys. 5.18. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie najlepszych z przebadanych kombinacji mutacji dla sterownika LSTM.

Tabela 5.8. Parametry algorytmu genetycznego dla eksperymentu 8 serii I

Nazwa parametru	Wartość
Rodzaj sterownika i liczba neuronów ukrytych	<i>LSTM - 8</i>
Rozmiar populacji ofiar i drapieżników	100
Początkowe wartości genów	$\mathcal{N}(0, 1)$
Początkowe wypełnienie genotypu	100%
Metoda selekcji	<i>rankingowa</i>
Liczność selekcji elitarnej	0
Metoda krzyżowania	<i>parametr badany</i>
Prawdopodobieństwo krzyżowania	<i>parametr badany</i>
Prawdopodobieństwo mutacji pojedynczego genu	<i>parametr badany</i>
Prawdopodobieństwo duplikacji genów	<i>parametr badany</i>
Prawdopodobieństwo wypełnienia genów	<i>parametr badany</i>
Prawdopodobieństwo usunięcia genów	<i>parametr badany</i>

### Eksperyment 8 - Badanie wpływu metody krzyżowania w algorytmie genetycznym na skuteczność ewolucji przy sterowniku wykorzystującym sieć LSTM

W niniejszym eksperymencie zostaną przeprowadzone badania, tych samych co w przypadku sterownika RNN, metod krzyżowania (opisanych w rozdziale 4.2.2) oraz prawdopodobieństwa jego zajścia dla najlepszych kombinacji mutacji uzyskanych w eksperymencie 7. Pozostałe parametry algorytmu genetycznego przedstawia tabela 5.8 Dla uproszczenia zapisu, zastosowano następujące oznaczenia:

- $SPp$  - krzyżowanie jedno-punktowe (*ang. Single-Point*),
- $TPp$  - krzyżowanie dwu-punktowe (*ang. Two-Point*),
- $SPPp$  - krzyżowanie jedno-punktowe na każdy segment (*ang. Single-Point per Part*),
- $TPPp$  - krzyżowanie dwu-punktowe na każdy segment (*ang. Two-Point per Part*),

gdzie  $p$  jest prawdopodobieństwem (w procentach) zajścia danego krzyżowania.

### Wyniki

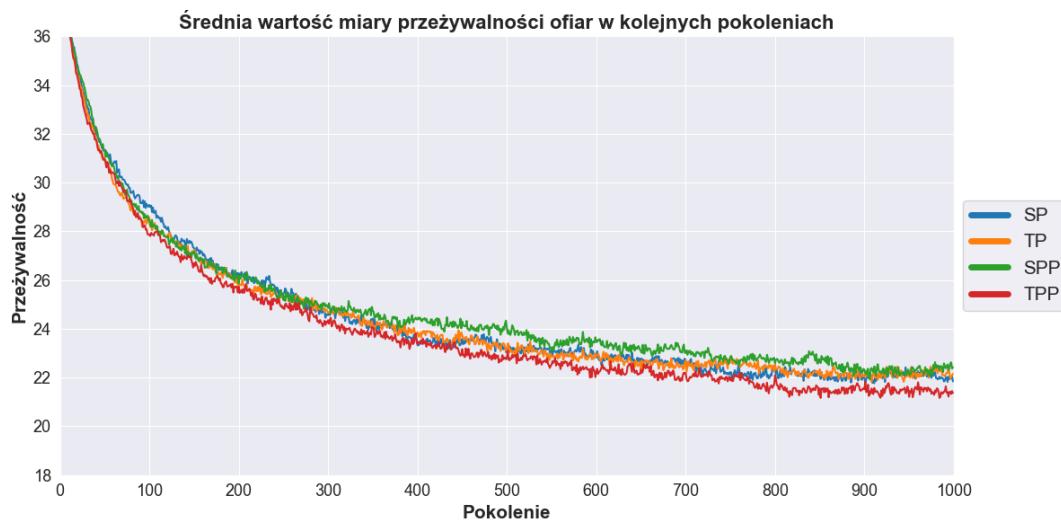
Wyniki eksperymentu zamieszczono na rysunkach 5.19 i 5.20, jako wykresy średniej miary przeżywalności ofiar na przestrzeni kolejnych pokoleń. Rysunek 5.19 przedstawia wyniki, które zostały uśrednione według poszczególnych kombinacji mutacji oraz wartości prawdopodobieństwa. Uśrednione rezultaty każdej z metod krzyżowania dla wartości prawdopodobieństwa wynoszącej [20%, 30%, 40%], przedstawia rysunek 5.19.

Najlepsze z uzyskanych wyników dla każdej z metod krzyżowania zamieszczono na rysunku 5.20. Dla porównania przedstawiono również wyniki bez krzyżowania.

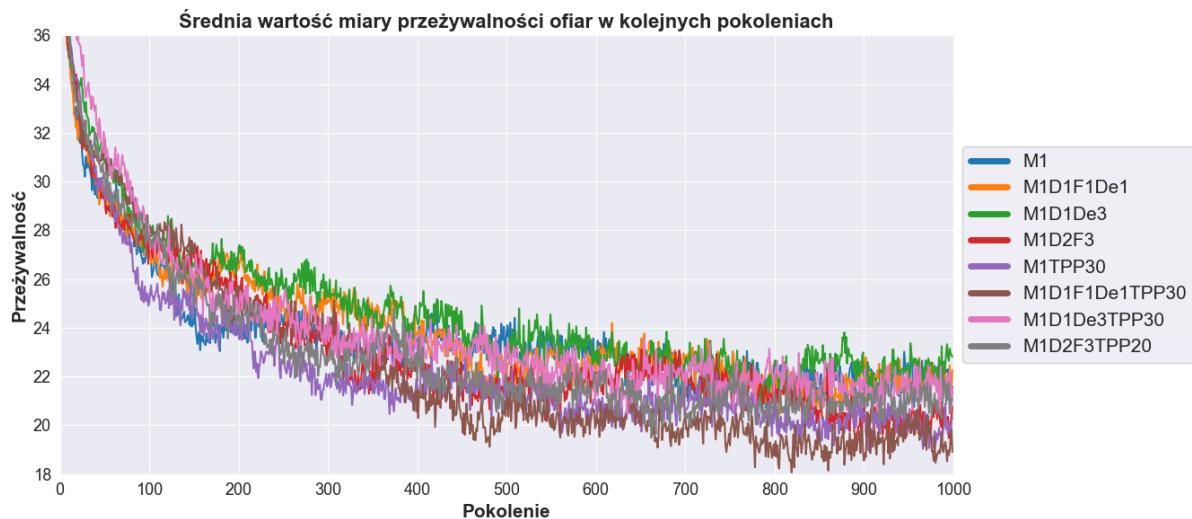
### Komentarz

W przypadku sterownika RNN, zbyt wysokie prawdopodobieństwa krzyżowania sprawdzały się kiepsko, dlatego przeprowadzono badania dla wartości, które dawały najlepsze rezultaty (20%, 30%, 40%). Na podstawie wyników można zauważyc, że wszystkie zbadane metody krzyżowania uzyskały dość zblżone rezultaty, jednak metoda  $TPP$  sprawdza się nieco lepiej od pozostałych (w przeciwieństwie do RNN, w którym wypadła zauważalnie najsłabiej). Ta różnica może wynikać z wewnętrznej budowy sieci oraz długości genotypu.

Poprawa skuteczności procesu ewolucji po zastosowaniu krzyżowania jest niewielka. Najlepiej sprawdziła się mutacja  $M1D1F1De1$  (tak jak w przypadku RNN) z krzyżowaniem  $TPP30$ , dlatego została wybrana do dalszych badań.



Rys. 5.19. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie metod krzyżowania dla sterownika LSTM.



Rys. 5.20. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie najlepszych z przebadanych kombinacji mutacji dla sterownika LSTM.

Tabela 5.9. Parametry algorytmu genetycznego dla eksperymentu 9 serii I

Nazwa parametru	Wartość
Rodzaj sterownika i liczba neuronów ukrytych	LSTM - 8
Rozmiar populacji ofiar i drapieżników	100
Początkowe wartości genów	parametr badany
Początkowe wypełnienie genotypu	parametr badany
Metoda selekcji	rankingowa
Liczność selekcji elitarnej	0
Metoda krzyżowania	dwu-punktowa na segment
Prawdopodobieństwo krzyżowania	30%
Prawdopodobieństwo mutacji pojedynczego genu	1%
Prawdopodobieństwo duplikacji genów	1%
Prawdopodobieństwo wypełnienia genów	1%
Prawdopodobieństwo usunięcia genów	1%

## Eksperyment 9 - Badanie wpływu inicjalizacji wartości genotypu w algorytmie genetycznym na skuteczność ewolucji przy sterowniku wykorzystującym sieć LSTM

Analogicznie jak w przypadku sterownika RNN, badaniom poddane zostały wartości parametru odchylenia standardowego  $\sigma^2$  dla rozkładu normalnego oraz stopień początkowego wypełnienia genotypu  $St_{init}$ , wartościami różnymi od zera (pochodzącymi z rozkładu  $\mathcal{N}(0, \sigma^2)$ ). Do tej pory, w tym celu wykorzystywano wartości pochodzące z rozkładu normalnego  $\mathcal{N}(0, 1)$ , a genotyp był nimi wypełniany w 100%. Pozostałe parametry przedstawia tabela 5.9

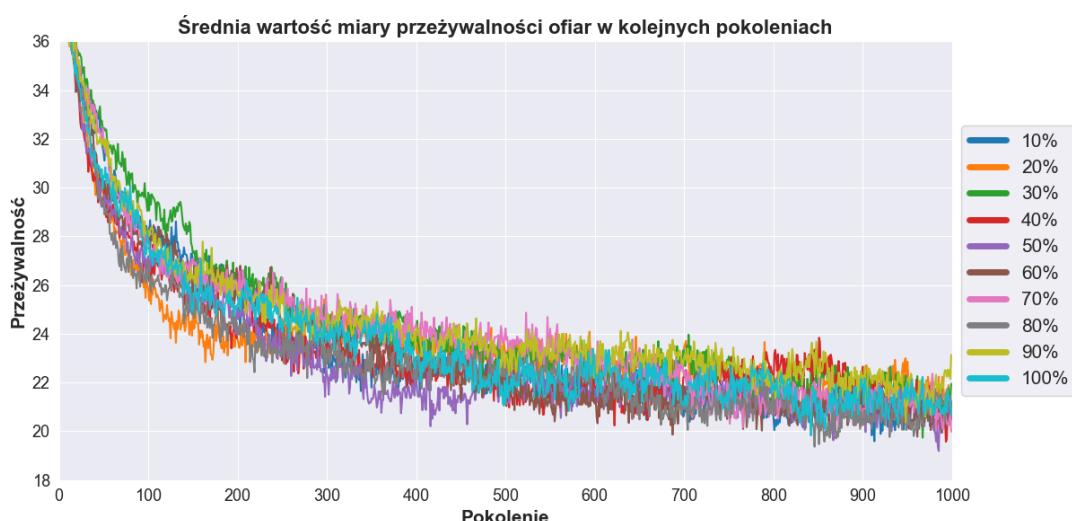
### Wyniki

Na rysunkach 5.21 i 5.22, przedstawiono miarę średniej przeżywalności ofiar w kolejnych pokoleniach, która obrazuje wyniki uzyskane w eksperymencie. Jako pierwszy badaniu poddano parametr  $St_{init}$  dla wartości 10% - 100% z krokiem co 10% (Rys. 5.22). Następnie, dla wybranego  $St_{init} = 80\%$ , przeprowadzono badania parametru  $\sigma^2$  dla wartości [0.5, 1, 2, 3, 5, 7]

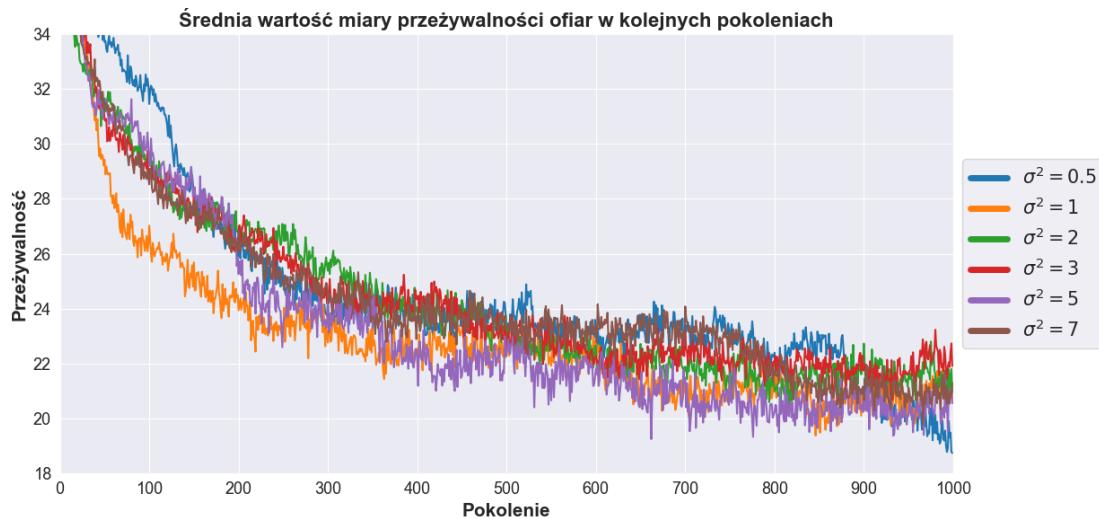
### Komentarz

Uzyskane rezultaty są znacznie bardziej zbliżone niż w przypadku sterownika RNN, co może świadczyć o mniejszej zależności sieci LSTM od początkowej inicjalizacji genotypu. Ostatecznie zdecydowano się na wybór wartości  $St_{init} = 80\%$ , ponieważ wyniki były nieco lepsze oraz charakteryzowały się większą powtarzalnością.

W przypadku parametru  $\sigma^2$ , uzyskane rezultaty są podobne, jednak można zauważyc przewagę wartości wykorzystywanej we wcześniejszych badaniach  $\sigma^2 = 1$ , dla której proces ewolucji w początkowych pokoleniach znacznie przebiegał szybciej. Ze względu na to, zdecydowano się na zastosowanie rozkładu  $\mathcal{N}(0, 1)$ .



Rys. 5.21. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie wartości początkowego wypełnienia  $St_{init}$  genotypu dla sterownika LSTM.



Rys. 5.22. Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie parametru  $\sigma^2$  z rozkładu normalnego dla sterownika LSTM.

### Eksperyment 10 - Badanie wpływu metody selekcji elitarnej w algorytmie genetycznym na skuteczność ewolucji przy sterowniku wykorzystującym sieć LSTM

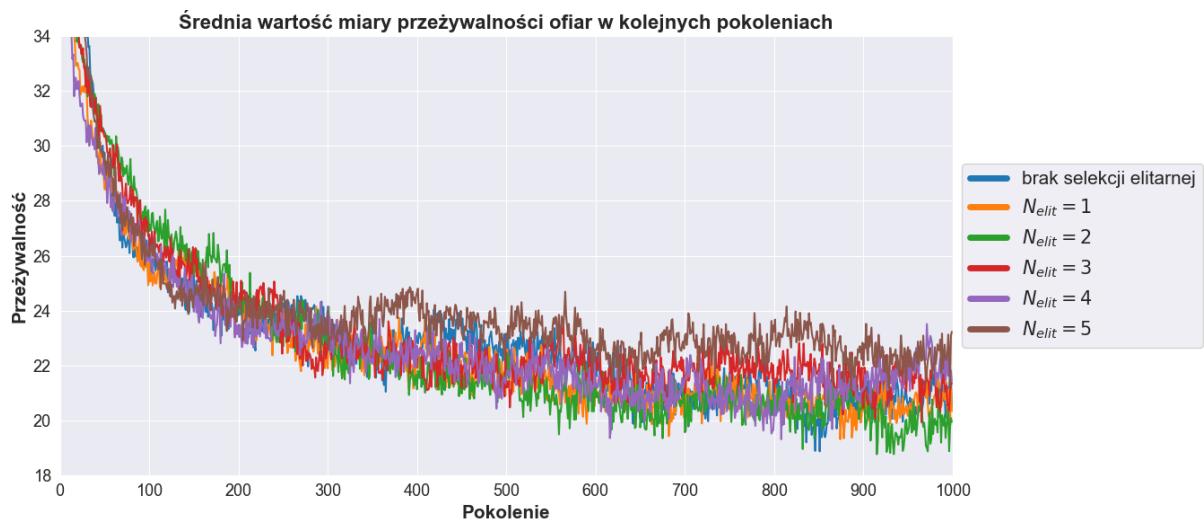
W ostatnim z eksperymentów dotyczących doboru parametrów, analogicznie jak dla sterownika RNN, przetestowane zostaną różne wartości parametru  $N_{elit}$  występującego w metodzie selekcji elitarnej.  $N_{elit}$  określa liczbę osobników, którzy w niezmiennej formie trafią do kolejnego pokolenia osobników. Wartości parametrów algorytmu genetycznego, ustalone według poprzednich eksperymentów przedstawia tabela 5.10.

Tabela 5.10. Parametry algorytmu genetycznego dla eksperymentu 10 serii I

Nazwa parametru	Wartość
Rodzaj sterownika i liczba neuronów ukrytych	LSTM - 8
Rozmiar populacji ofiar i drapieżników	100
Początkowe wartości genów	$\mathcal{N}(0, 1)$
Początkowe wypełnienie genotypu	80%
Metoda selekcji	rankingowa
Liczność selekcji elitarnej	parametr badany
Metoda krzyżowania	dwu-punktowa na segment
Prawdopodobieństwo krzyżowania	30%
Prawdopodobieństwo mutacji pojedynczego genu	1%
Prawdopodobieństwo duplikacji genów	1%
Prawdopodobieństwo wypełnienia genów	1%
Prawdopodobieństwo usunięcia genów	1%

### Wyniki

Uzyskane z eksperymentu wyniki zawiera rysunek 5.23. Przedstawiono na nim miarę średniej przeżywalności ofiar w zależności od wartości  $N_{elit}$ . Parametr  $N_{elit}$  został przebadany dla wartości [1, 2, 3, 4, 5] oraz dodatkowo, w celach porównawczych, zamieszczono rezultaty uzyskane bez wykorzystania metody selekcji elitarnej.



Rys. 5.23. Wykres średniej przeżywalności ofiar w kolejnych generacjach.  
Porównanie rozmiaru selekcji elitarnej  $N_{elit}$  dla sterownika LSTM.

### Komentarz

Podobnie jak w przypadku sterownika sieci RNN, nie zaobserwowano znaczającej poprawy skuteczności przebiegu ewolucji, a w niektórych przypadkach została ona wręcz pogorszona, jednak różnice są dość niewielkie i mogą być dziełem przypadku (wyniki pochodzą tylko z 5 przebiegów). Dla  $N_{elit} = 2$  pod koniec procesu uczenia uzyskano nieco lepsze wyniki, dlatego zdecydowano się na wykorzystanie w algorytmie genetycznym selekcji elitarnej o wartości 2.

### 5.1.3. PODSUMOWANIE BADAŃ - WYBÓR WARTOŚCI PARAMETRÓW ALGORYTMU GENETYCZNEGO

Pomimo znacznych różnic w budowie wewnętrznej oraz reprezentacji genotypowej (zwłaszcza jego długości) między siecią RNN oraz LSTM, wyniki uzyskane w badaniach doboru parametrów uczenia są dość zbliżone. Ostatecznie wybrane parametry algorytmu genetycznego, które zostaną zastosowane w dalszych badaniach przedstawia tabela 5.11.

Tabela 5.11. Ostatecznie wybrane parametry algorytmu genetycznego dla obydwu sterowników.

Nazwa parametru	Wartości dla RNN	Wartości dla LSTM
Rozmiar populacji ofiar i drapieżników	100	100
Początkowe wartości genów	$\mathcal{N}(0, 2)$	$\mathcal{N}(0, 1)$
Początkowe wypełnienie genotypu	30%	80%
Metoda selekcji	<i>rankingowa</i>	<i>rankingowa</i>
Liczność selekcji elitarnej	2	2
Metoda krzyżowania	<i>jedno-punktowa</i>	<i>dwu-punktowa na segment</i>
Prawdopodobieństwo krzyżowania	30%	30%
Prawdopodobieństwo mutacji pojedynczego genu	1%	1%
Prawdopodobieństwo duplikacji genów	1%	1%
Prawdopodobieństwo wypełnienia genów	1%	1%
Prawdopodobieństwo usunięcia genów	1%	1%

## **5.2. SERIA II - BADANIE WPŁYWU LICZBY NEURONÓW UKRYTYCH STEROWNIKA OFIAR NA TWORZENIE SIĘ ZACHOWAŃ STADNYCH PRZY PRESJI ATAKÓW DRAPIEŻNIKA**

Liczba neuronów ukrytych jest jednym z najważniejszych hiperparametrów sieci neuronowej. W przypadku sterownika wyposażonego w rekurencyjną sieć (RNN lub LSTM), odpowiednio dobrana może zwiększyć zdolności agentów i przyczynić się do wykształcenia bardziej złożonych zachowań. Dzięki temu, agenty będą miały możliwość wykorzystania dodatkowych zdolności, takich jak mechanizm komunikacji. Przy niewłaściwym doborze, zbyt wysoka liczba neuronów może bardzo spowolnić proces nauki drogą ewolucji, a zbyt mała nawet całkowicie go uniemożliwić. W poprzednich badaniach doboru parametrów algorytmu genetycznego, obydwa sterowniki wyposażone były w 8 neuronów ukrytych (wartość stosowana w pracy [12]).

W eksperymentach 1 - 4, serii II, przeprowadzono badania mające na celu sprawdzenie w jaki sposób liczba neuronów sterownika agentów ofiar, wpływa na tworzenie się wśród nich zachowań stadnych, a także wybór wartości oferującej najlepsze możliwości rozwoju agentów. Dodatkowo, badania zostaną wykonane kolejno przy wyłączonym oraz włączonym mechanizmie komunikacji ofiar

Do badań wykorzystano wcześniej wyselekcjonowany genotyp drapieżnika, pochodzący od jednostki, która charakteryzowała się najwyższą skutecznością ataków. Można go znaleźć w pliku *predator\_rnn\_genes.pkl* (dla sterownika RNN) oraz *predator\_lstm\_genes.pkl* (dla sterownika LSTM), na załączonej do pracy płycie DVD. Populacja drapieżników składa się z dwóch osobników oraz została wyłączona z procesu dalszej ewolucji. Dzięki tym zabiegom, porównanie wyników jest łatwiejsze i bardziej miarodajne. Symulacja rozpoczyna się od fazy rozgrzewki trwającej 250 kroków, w której drapieżnicy są jeszcze nieobecni, co ma dać populacji ofiar czas niezbędny na przygotowanie się do ataków drapieżnika. Takie rozwiązanie zastosowano również w pracach [12] i [16].

Wszystkie wartości parametrów środowiska symulacyjnego zawarto w tabeli 4.4. Część z nich, oznaczona jako "*parametr dostrajany*", jest dostosowywana według potrzeb poszczególnych badań. Parametry środowiska przyjęte na potrzeby eksperymentów z serii II zawiera tabela 5.12. Wartości parametrów algorytmu genetycznego przedstawia tabela 5.11.

Tabela 5.12. Wartości parametrów środowiska dla eksperymentów 1 - 4, serii II

<b>Nazwa parametru</b>	<b>Wartość</b>
<i>Parametry symulacji</i>	
Rozmiar mapy	512 wm x 512 wm
Liczba pokoleń	1000
Liczba kroków symulacji	2000
Początkowa liczba kroków bez drapieżnika	250
<i>Parametry agentów ofiary</i>	
Liczba kopii agenta na początku symulacji	50
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	4 / 6 / 8 / 12 / 16
<i>Parametry agentów drapieżnika</i>	
Liczba kopii agenta w symulacji	2
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	8

## **5.2.1. BADANIA DLA STEROWNIKA WYKORZYSTUJĄCEGO SIEĆ RNN**

### **Eksperyment 1 - Badanie wpływu liczby ukrytych neuronów na przeżywalność oraz formowanie się zachowań stadnych wśród ofiar przy presji ataków drapieżnika dla sterownika wykorzystującego sieć RNN**

W eksperymencie wykorzystano miarę średniej przeżywalności ofiar i liczby prób ataków drapieżnika, a także gęstości oraz rozpiętości stada ofiar. Sterownikiem obydwu rodzajów agentów (ofiar i drapieżników) jest sieć RNN. Mechanizm komunikacji ofiar jest wyłączony. Drapieżnicy zostali wyposażeni we wcześniej wybrany genotyp oraz wyłączeni z procesu dalszej ewolucji.

#### **Wyniki**

Wyniki eksperymentu zostały zobrazowane na rysunkach 5.24 - 5.27, przedstawiających wykresy średniej wartości wykorzystanych miar w kolejnych pokoleniach. W kolorze niebieskim, pomarańczowym, zielonym, czerwonym i fioletowym, przedstawiono wyniki uzyskane dla sterownika wyposażonego w kolejno 4, 6, 8, 12 i 16 ukrytych neuronów.

#### **Komentarz**

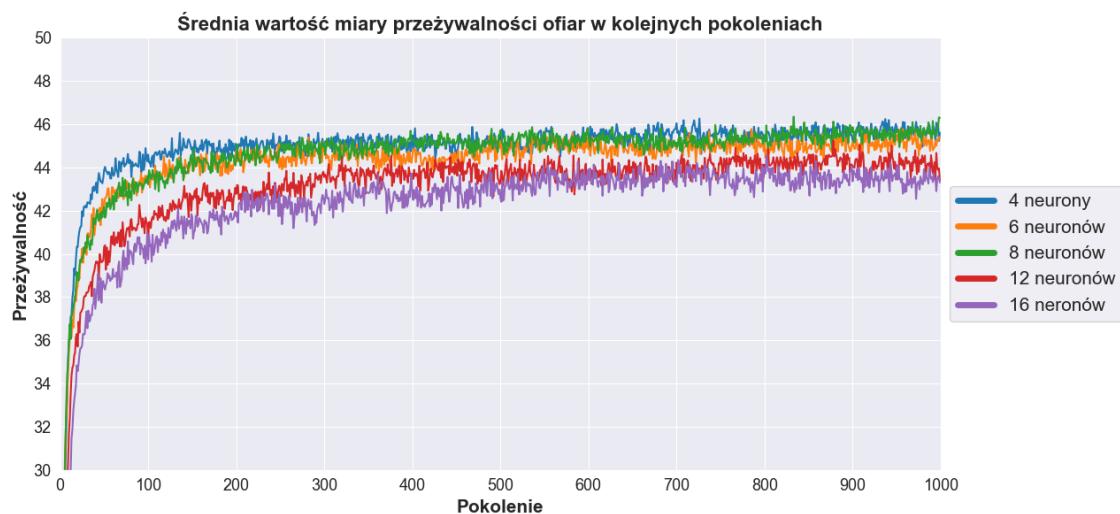
Na podstawie uzyskanych wyników można zauważyć, że przy mniejszej liczbie neuronów ukrytych sterownika (4, 6, 8), średnia przeżywalność ofiar jest dość podobna, natomiast spada wraz ze wzrostem liczby neuronów (12, 16).

Przyczyną tego może być wzrost rozmiaru genotypu, proporcjonalny do liczby neuronów sterownika, a tym samym liczby genów, które należy właściwie dopasować, dlatego proces ewolucji może zachodzić mniej skutecznie. W dodatku, badania parametrów algorytmu genetycznego były przeprowadzane dla sterownika wyposażonego w 8 neuronów, co również mogło mieć wpływ na uzyskane rezultaty. Zastosowanie innych parametrów uczenia, w przypadku wyższych wartości liczby neuronów, mogłoby skutkować uzyskaniem lepszych wyników.

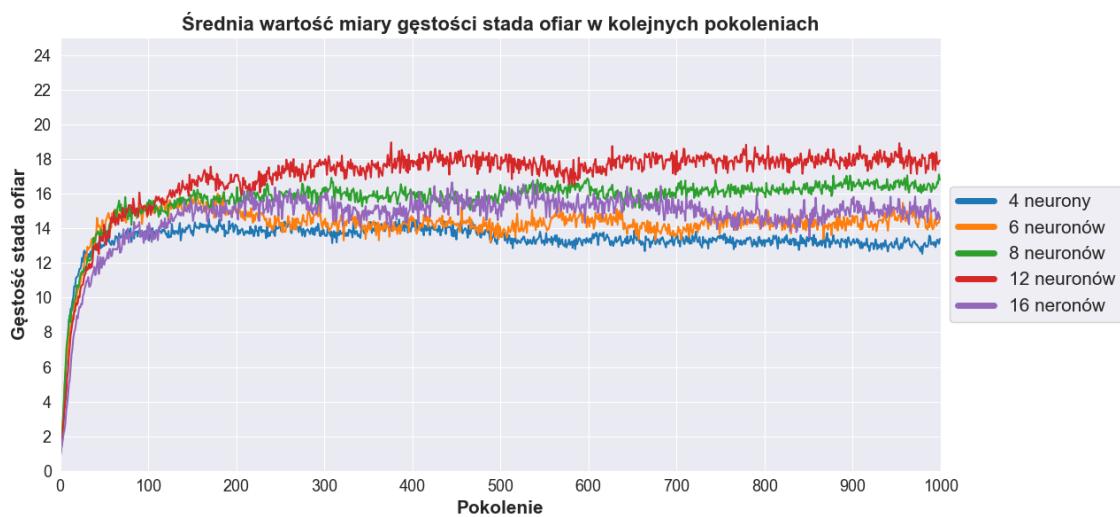
Porównując wykresy znajdujące się na rysunkach 5.24 oraz 5.25, można zaobserwować, że w początkowej fazie ewolucji, zwiększenie gęstości stada ofiar wpływa pozytywnie na ich przeżywalność, jednakże po przekroczeniu pewnej wartości, dalszy wzrost gęstość stada nie jest równoznaczny z jeszcze wyższą przeżywalnością. W przypadku miary rozpiętości stada (Rys. 5.26), wraz ze spadkiem jej wartości, wzrost przeżywalności ofiar zachodzi sukcesywnie. Dodatkowo, przy zastosowaniu mniejszej liczby neuronów, rozpiętość stada również maleje, co odpowiada wzrostowi miary przeżywalności. Wynika z tego, że wzrost przeżywalności ofiar jest bardziej powiązany z rozpiętością stada, czyli średnią minimalną odległością między ofiarami, niż z jego gęstością.

Liczba prób ataków drapieżnika jest dość zbliżona w każdym przypadku, jednak zmniejsza się zauważalnie wolniej na początkowym etapie procesu ewolucji przy liczbie neuronów wynoszących 12 i 16.

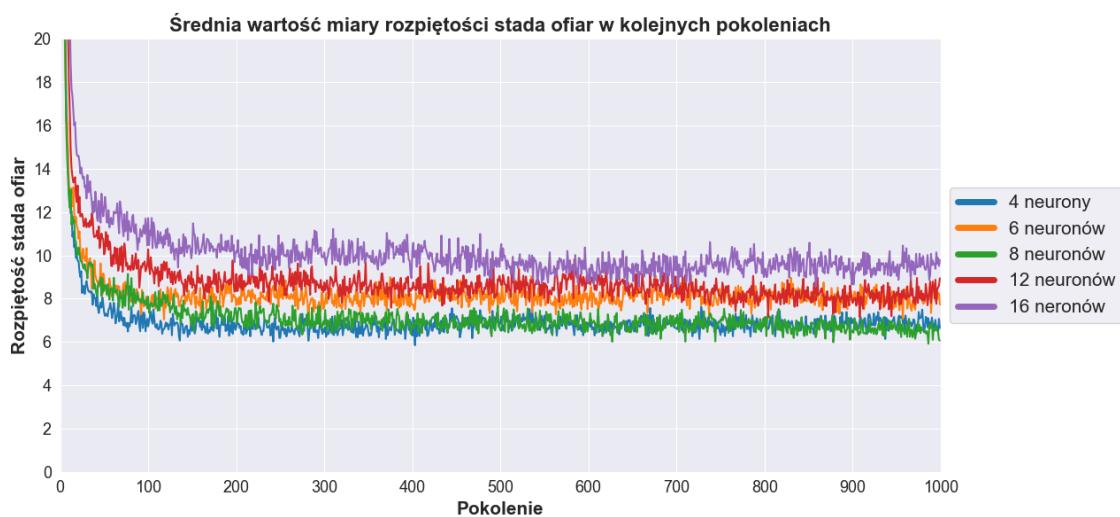
Biorąc pod uwagę uzyskane wyniki, najlepiej wypadł sterownik z 8 neuronami. Osiąga najwyższe wartości miary przeżywalności, porównywane z liczbą neuronów 4 i 6, zbliżoną do nich rozpiętość stada, a nawet wyższe wartości gęstości stada. Oferuje przy tym również większe możliwości rozwoju agentów.



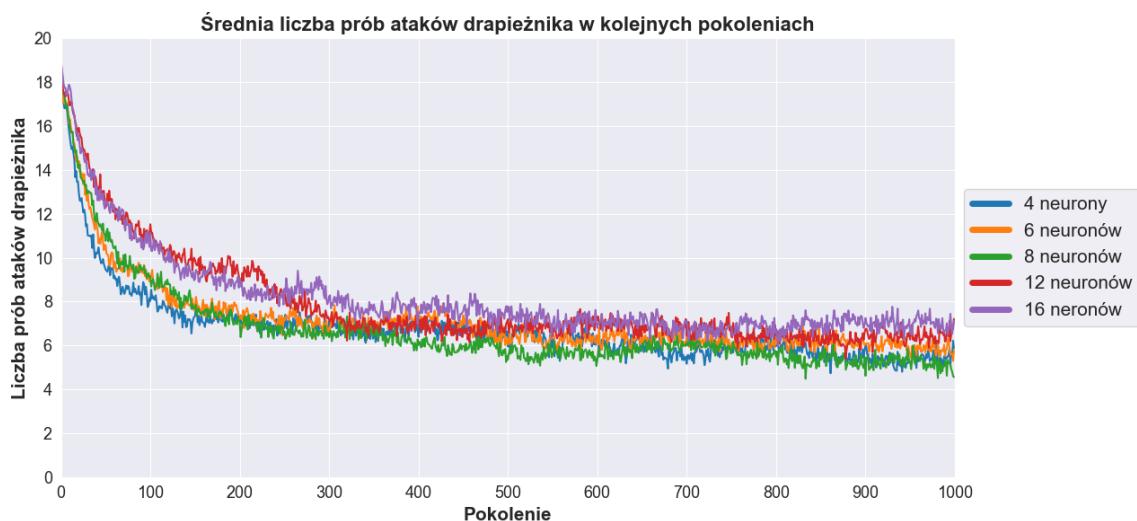
Rys. 5.24. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika RNN.



Rys. 5.25. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika RNN.



Rys. 5.26. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika RNN.



Rys. 5.27. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika RNN.

## Eksperyment 2 - Badanie wpływu liczby ukrytych neuronów na przeżywalność oraz formowanie się zachowań stadnych wśród ofiar przyłączonym mechanizmie komunikacji ofiar oraz presji ataków drapieżnika dla sterownika wykorzystującego sieć RNN

W eksperymencie wykorzystano miarę średniej przeżywalności ofiar i liczby prób ataków drapieżnika, a także gęstości oraz rozpiętości stada ofiar. Sterownikiem obydwu rodzajów agentów (ofiar i drapieżników) jest sieć RNN. Komunikacja ofiar została w tym przypadku włączona. Drapieżnikom przypisano wcześniej wyselekcjonowany genotyp. Zostali również wyłączeni z procesu ewolucji.

### Wyniki

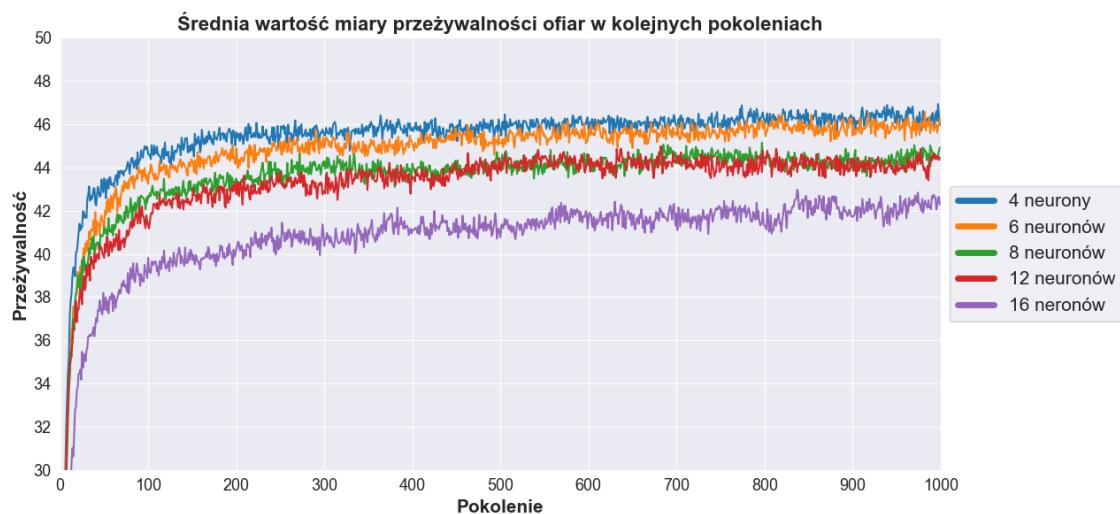
Wyniki eksperymentu przedstawiono na rysunkach 5.28 - 5.31, które zawierają wykresy średniej wartości wykorzystanych miar. W kolorze niebieskim, pomarańczowym, zielonym, czerwonym i fioletowym, przedstawiono wyniki uzyskane dla sterownika wyposażonego w kolejno 4, 6, 8, 12 i 16 ukrytych neuronów.

### Komentarz

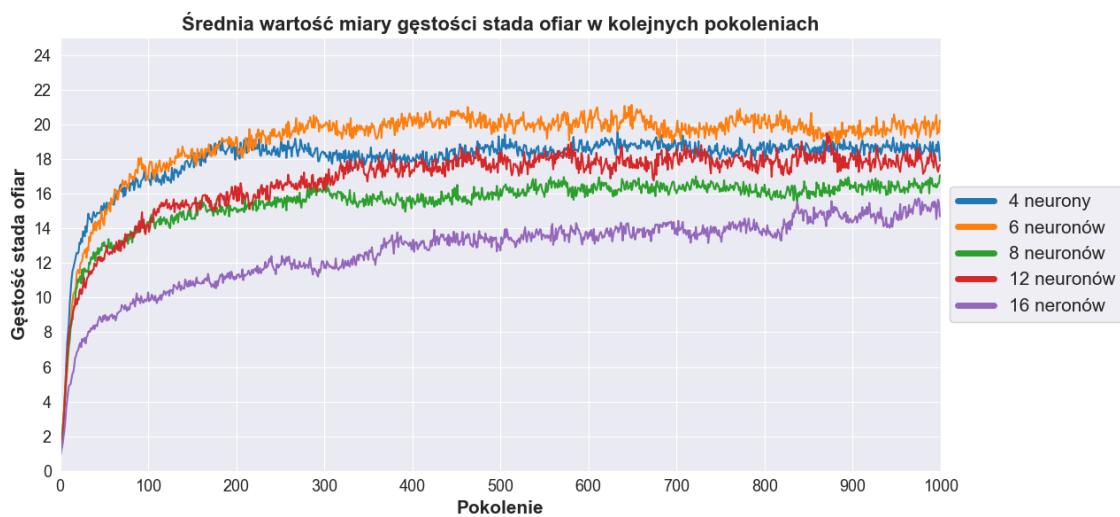
Wyniki uzyskane przyłączonym mechanizmie komunikacji ofiar są podobne do tych otrzymanych w eksperymencie 1. Można jednak zauważać, że różnice w wartościach wykorzystanych miar dla poszczególnych liczb neuronów ukrytych, są jeszcze większe.

Wprowadzenie mechanizmu komunikacji spowodowało również ogólny wzrost średniej liczby prób ataków drapieżnika i można zauważać, że zwiększa się dodatkowo wraz z liczbą neuronów sterownika. Liczba prób ataków jest również powiązana z miarą rozpiętości stada. Im większa jest jej wartość, a poszczególne ofiary są od siebie bardziej oddalone, tym więcej prób ataków są w stanie przeprowadzić drapieżnicy.

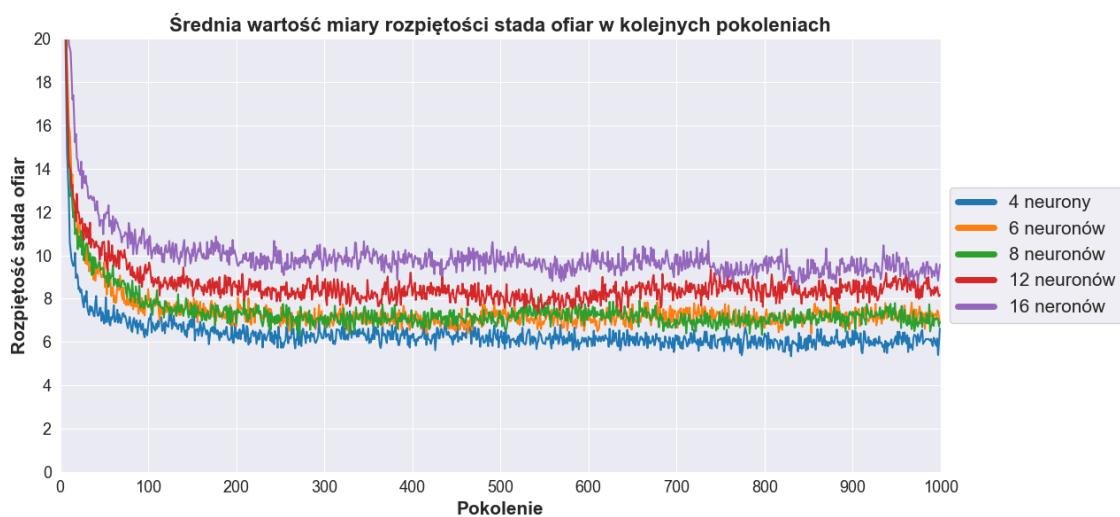
Uzyskane wyniki pokazują, że skuteczność procesu ewolucji była znacznie mniejsza w przypadku agentów wyposażonych w sterownik o liczbie neuronów równej 16. Takie ofiary cechowały się znacznie mniejszą przeżywalnością, formowane stada były mniej gęste oraz bardziej rozpięte, a przede wszystkim, liczba prób ataków drapieżnika była aż dwukrotnie większa niż dla sterownika o 4 i 6 neuronach.



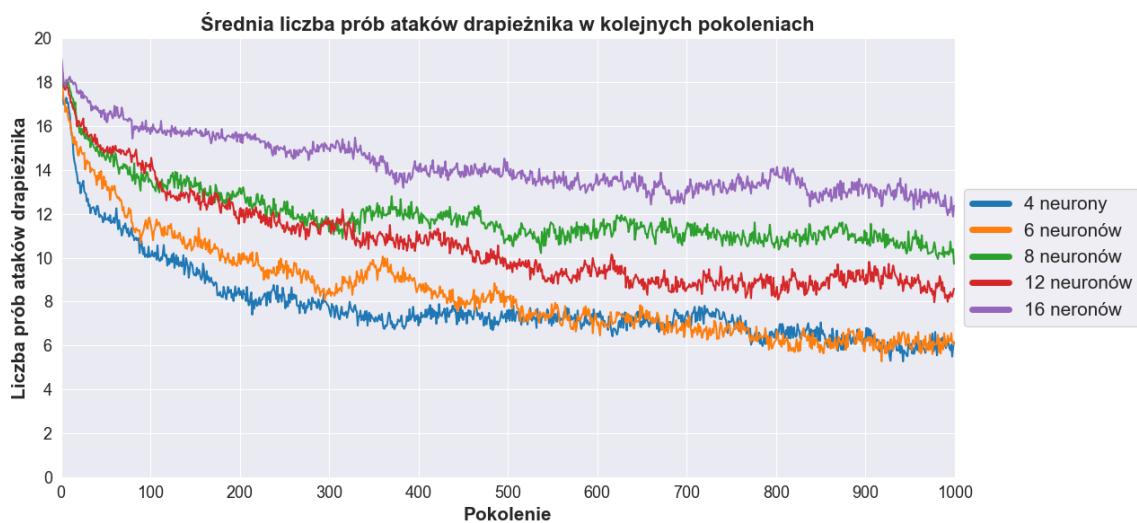
Rys. 5.28. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika RNN.



Rys. 5.29. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika RNN.



Rys. 5.30. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika RNN.



Rys. 5.31. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika RNN.

## 5.2.2. BADANIA DLA STEROWNIKA WYKORZYSTUJĄCEGO SIEĆ LSTM

### Eksperyment 3 - Badanie wpływu liczby ukrytych neuronów na przeżywalność oraz formowanie się zachowań stadnych wśród ofiar przy presji ataków drapieżnika dla sterownika wykorzystującego sieć LSTM

Do badań wykorzystano miarę średniej przeżywalności ofiar, liczby prób ataków drapieżnika oraz gęstości i rozpiętości stada ofiar. Jako sterownik obydwu agentów zastosowano sieć LSTM. Mechanizm komunikacji ofiar pozostał wyłączony. Drapieżnicy zostali wyposażeni we wcześniej wybrany genotyp oraz wyłączeni z procesu dalszej ewolucji.

#### Wyniki

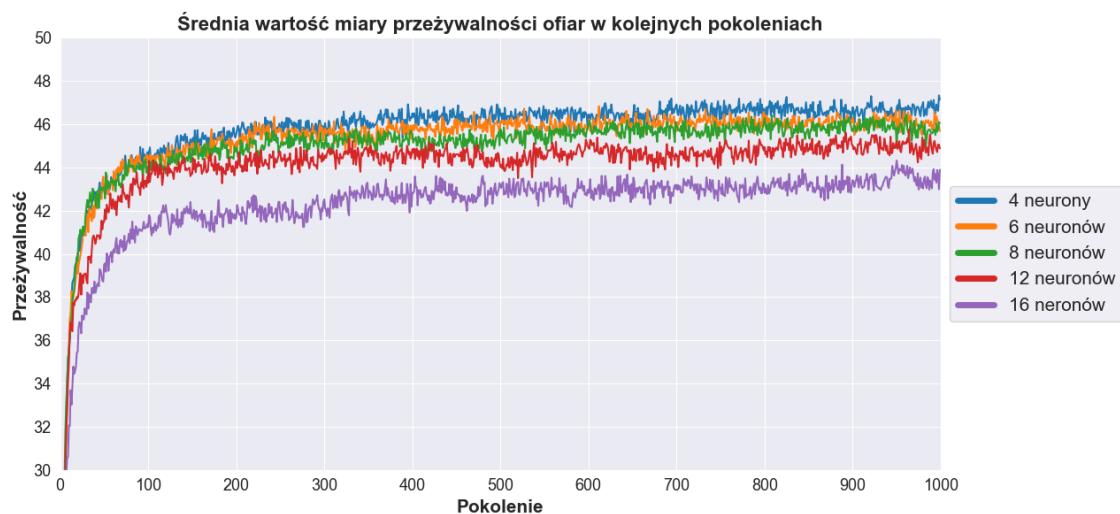
Wyniki eksperymentu zaprezentowano na rysunkach 5.32 - 5.35, przedstawiających wykresy średniej wartości wykorzystanych miar dla kolejnych pokoleń. Kolorem niebieskim, pomarańczowym, zielonym, czerwonym i fioletowym, przedstawiono wyniki uzyskane dla sterownika sieci LSTM, posiadającego kolejno 4, 6, 8, 12 i 16 neuronów ukrytych.

#### Komentarz

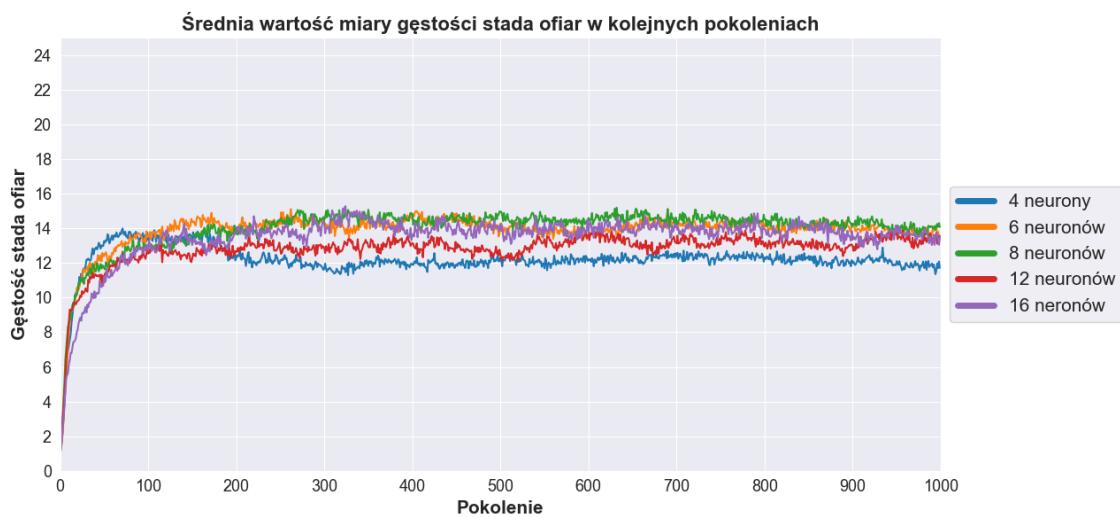
Podobnie jak w przypadku sterownika RNN, wraz ze wzrostem liczby neuronów ukrytych zaobserwowano spadek miary średniej przeżywalności ofiar. Wynika z tego, że dla agentów wyposażonych w sterownik o mniejszej liczbie neuronów, a tym samym mniejszy genotyp, proces ewolucji zachodzi skuteczniej i umożliwia łatwiejsze wykształcenie zachowań pozwalających na przetrwanie w środowisku.

Wykresy miar średniej gęstości oraz rozpiętości stada ofiar dla poszczególnych wartości liczby neuronów, są do siebie bardziej zbliżone, niż miało to miejsce w przypadku sterownika RNN. Może to świadczyć o tym, że skuteczność procesu ewolucji dla sterownika wyposażonego w sieć LSTM jest mniej zależna od zastosowanej liczby neuronów ukrytych.

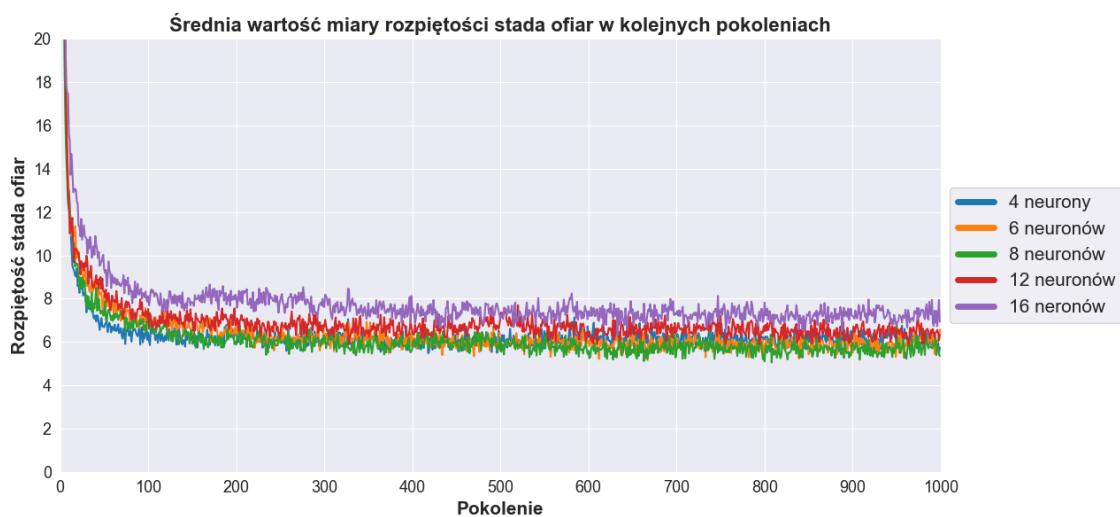
Średnia liczba ataków wzrasta wraz z liczbą neuronów, co zostało zaobserwowane również w poprzednich eksperymentach, jednak dla 16 neuronów, wyniki są znaczco wyższe od pozostałych, co może być spowodowane większą rozpiętością stada.



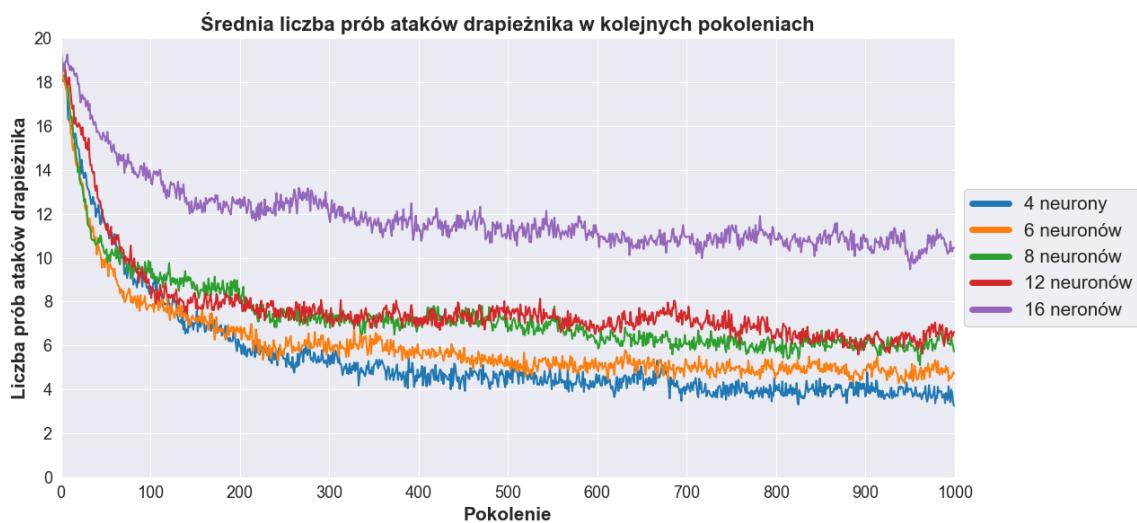
Rys. 5.32. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika LSTM.



Rys. 5.33. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika LSTM.



Rys. 5.34. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika LSTM.



Rys. 5.35. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika LSTM.

#### Eksperyment 4 - Badanie wpływu liczby ukrytych neuronów na przeżywalność oraz formowanie się zachowań stadnych wśród ofiar przyłączonym mechanizmie komunikacji ofiar oraz presji ataków drapieżnika dla sterownika wykorzystującego sieć LSTM

W eksperymencie została wykorzystana miara średniej przeżywalności ofiar, liczby prób ataków drapieżnika oraz miara gęstości i rozpiętości stada ofiar. Sterownikiem obydwu agentów jest sieć LSTM. Mechanizm komunikacji ofiar jest w tym przypadku włączony. Drapieżników wyposażono we wcześniej wybrany genotyp i wykluczono z procesu dalszej ewolucji.

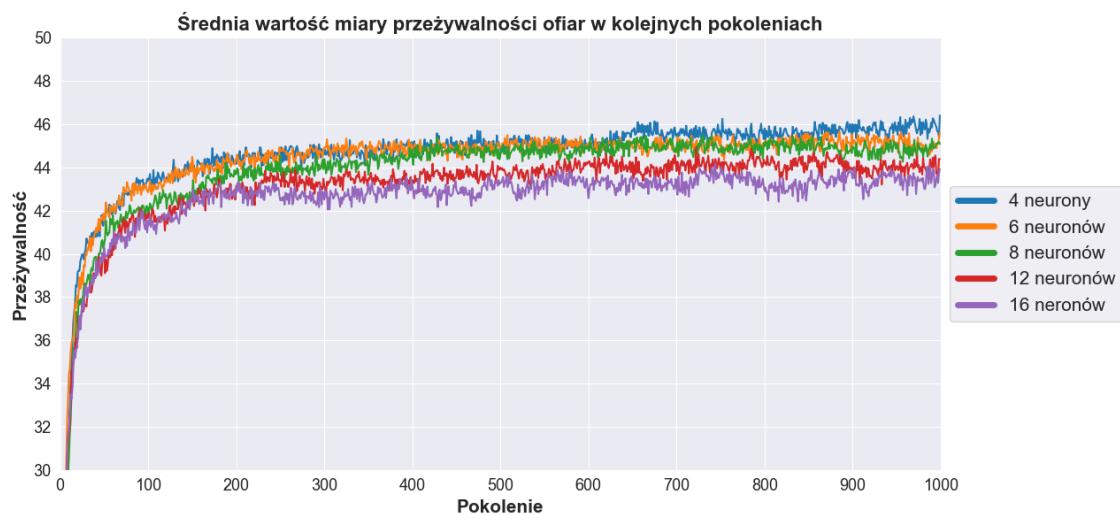
#### Wyniki

Wyniki eksperymentu przedstawione zostały na rysunkach 5.36 - 5.39, które zawierają wykresy średniej wartości każdej z wykorzystanych miar dla kolejnych pokoleń. W kolorze niebieskim, pomarańczowym, zielonym, czerwonym i fioletowym, przedstawiono wyniki otrzymane kolejno dla 4, 6, 8, 12 i 16 neuronów ukrytych sterownika LSTM.

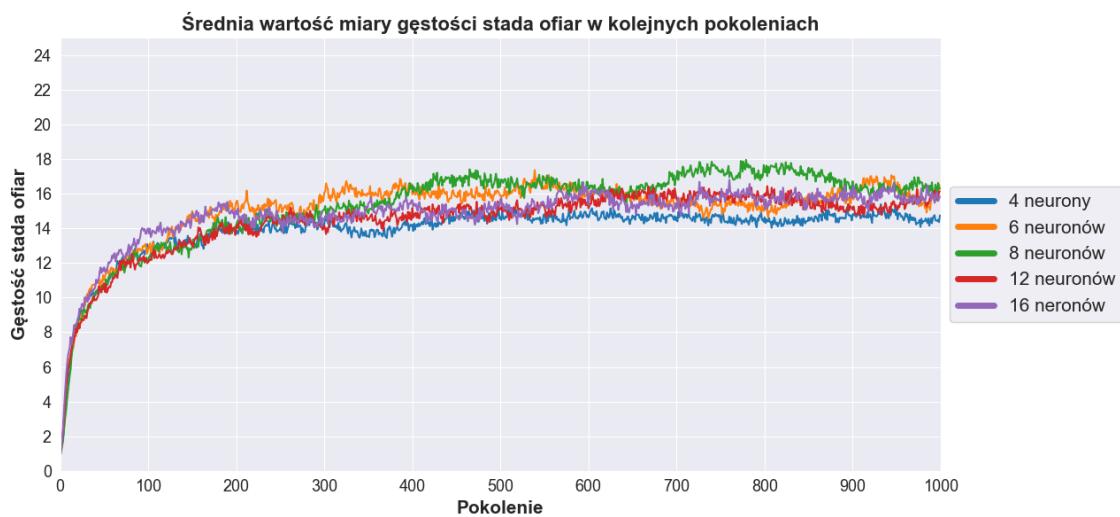
#### Komentarz

Średnia przeżywalność ofiar nie uległa poprawie po zastosowaniu mechanizmu komunikacji, jednak jej wartości osiągane dla poszczególnych liczb neuronów, stały się dzięki temu bardziej zbliżone (w przeciwnieństwie do sieci RNN, w której wprowadzenie mechanizmu komunikacji jeszcze bardziej uwidocznioło te różnice).

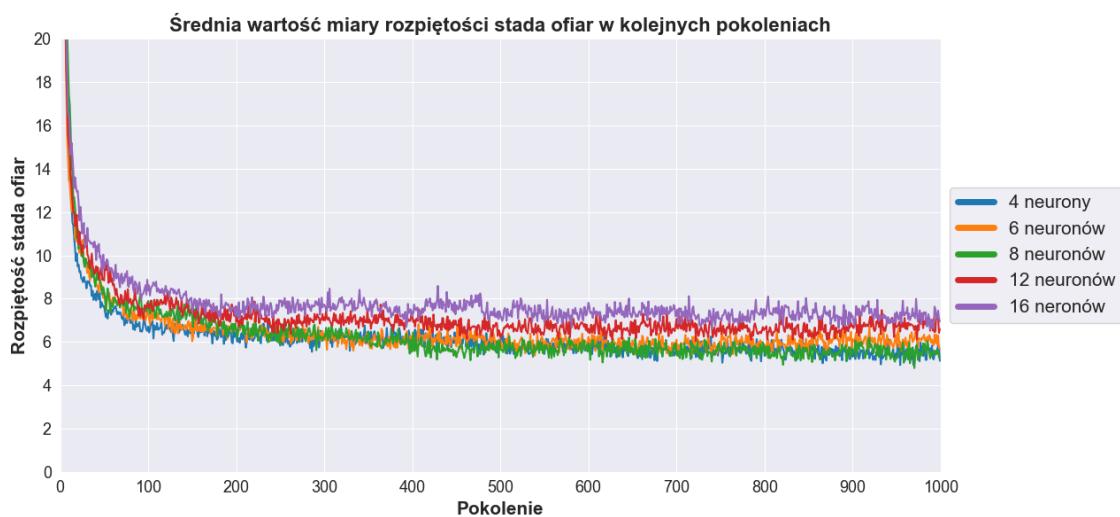
Wykresy wartości pozostałych miar wyglądają podobnie, jednak można zauważyć, że niezależnie od zastosowanej liczby neuronów, miara prób ataków drapieżników znaczaco wzrosła w stosunku do przypadku bez mechanizmu komunikacji. Taka obserwacja miała miejsce również dla sterownika RNN. Zostało to dokładnie przedstawione i omówione w następnej sekcji, dotyczącej porównania uzyskanych wyników.



Rys. 5.36. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika LSTM.



Rys. 5.37. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika LSTM.



Rys. 5.38. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie liczby ukrytych neuronów sterownika LSTM.



Rys. 5.39. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika LSTM.

### 5.2.3. PORÓWNANIE WYNIKÓW EKSPERYMENTÓW SERII II

Wyniki uzyskane w ramach eksperymentów 1 - 4, zostały uśrednione względem liczby neuronów ukrytych oraz przedstawione na rysunkach 5.40 - 5.43 w celu porównania.

Wprowadzenie mechanizmu komunikacji ofiar nie wpłynęło pozytywnie na ich przeżywalność dla żadnego ze sterowników. Zaobserwowano natomiast znaczny wzrost gęstości stada. Stanowi to sytuację przeciwną do zaobserwowanej w pracy [12], w której mechanizmu komunikacji wywołał spadek gęstości formowanych stad, przy zachowaniu poziomu przeżywalności ofiar. Według wyjaśnień, zdolność wydawania dźwięków pozwoliła ostrzegać pobliskie ofiary przed niebezpieczeństwem, a tym samym formowanie tak ścisłych stad nie było już konieczne. Ten odwrotny efekt może wynikać z różnic w budowie środowiska badawczego, wykorzystanego na potrzeby niniejszej pracy, takich jak obecność dwóch drapieżników czy automatyczne wydawanie dźwięku przez ofiarę, kiedy w polu jej widzenia pojawi się drapieżnik (w pracy [12] ofiary same decydowały o wydaniu dźwięku).

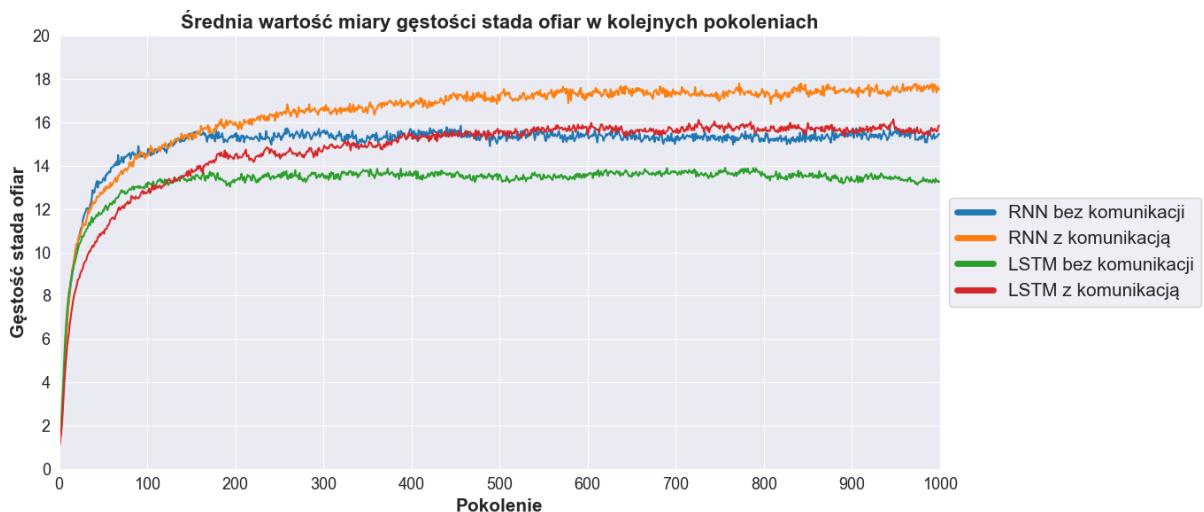
Bez mechanizmu komunikacji ofiar, miara prób ataków drapieżnika dla obydwu sterowników była na tym samym poziomie. Co ciekawe, po jego wprowadzeniu, liczba prób ataków znacznie wzrosła. Może to wynikać z tego, że gdy ofiara wyda dźwięk ostrzegawczy, informujący o obecności drapieżnika, pobliskie osobniki przybędą jej z pomocą, narażając się tym samym na atak, który jednak dzięki obecności większej liczby ofiar, z dużym prawdopodobieństwem nie będzie udany. To zjawisko tłumaczyłoby również większą wartość miary gęstości stada.

Na podstawie wykresów gęstości oraz rozpiętości (Rys. 5.41 i 5.42) można wnioskować, że niezależnie od mechanizmu komunikacji, ofiary sterowane siecią LSTM preferują formowanie bardziej ścisłych (o niższej rozpiętości), lecz mniej gęstych stad, natomiast w przypadku sieci RNN, tworzone stada są bardziej gęste, jednak poszczególne osobniki znajdują się w nieco większej odległości od siebie (większa rozpiętość).

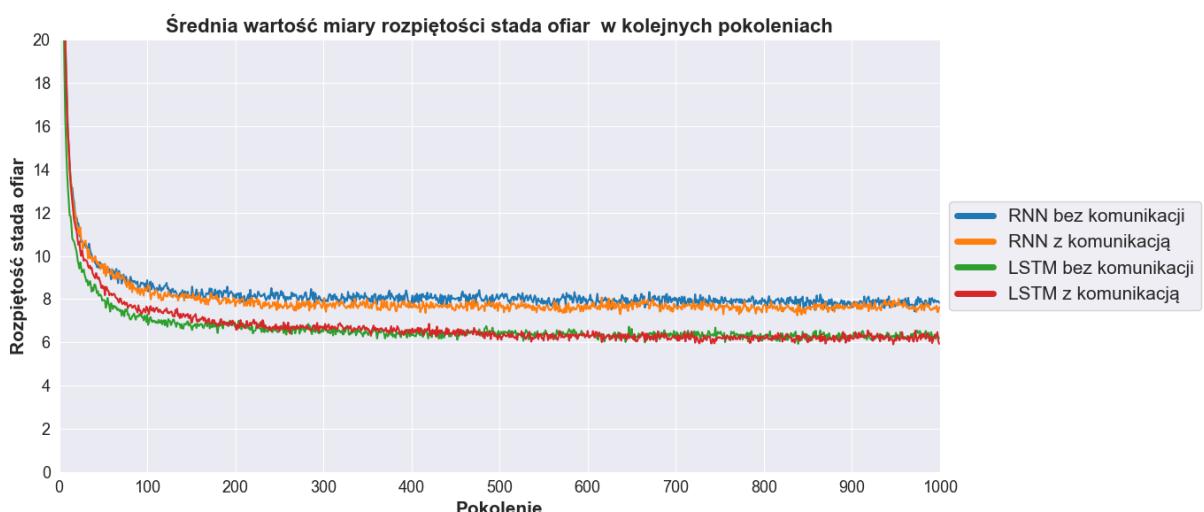
Słowem podsumowania, wyniki eksperymentów przeprowadzone w ramach serii I pokazują, że ofiary skutecznie nauczyły się wykorzystywać ograniczenie drapieżników (mechanizm dezorientacji) na swoją korzyść i wykształciły zachowanie grupowania się, umożliwiające przetrwanie w większej liczności. Wprowadzenie mechanizmu komunikacji wpłynęło na wzrost gęstości formowanych stad, jednak nie przełożyło się to na poprawę przeżywalności. Co ciekawe, mechanizm komunikacji spowodował znaczny wzrost liczby prób ataków drapieżnika, jednak nie wpłynęło to na przeżywalność ofiar (ataków było więcej, lecz nie były one skuteczne).



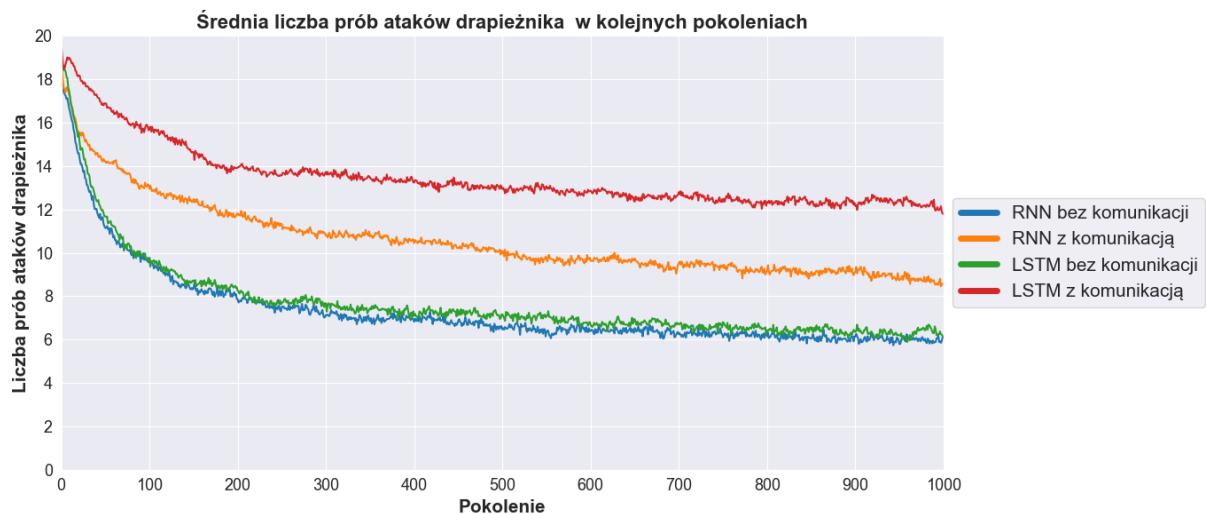
Rys. 5.40. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników.



Rys. 5.41. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników.



Rys. 5.42. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników.



Rys. 5.43. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.  
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników.

### 5.3. SERIA III - WPŁYW MECHANIZMU KOMUNIKACJI OFIAR NA TWORZENIE SIĘ ZACHOWAŃ STADNYCH PRZY PRESJI SPOWODOWANEJ OBECNOŚCIĄ WIELU DRAPIEŻNIKÓW

W badaniach przeprowadzonych w ramach prac [15, 16], wzięto pod uwagę obecność wyłącznie jednego drapieżnika w środowisku. W takich warunkach, ofiary wykorzystując mechanizm dezorientacji na swoją korzyść, były w stanie skutecznie przeciwdziałać jego atakom, dzięki formowaniu stad. Mechanizm komunikacji ofiar, zastosowany i poddany badaniom w pracy [12], dodatkowo to ułatwił, ponieważ zyskały możliwość poinformowania pobliskich osobników o niebezpieczeństwie (obecności drapieżnika), co pozwoliło skuteczniej go unikać. Wprowadzenie do środowiska wielu drapieżników może mieć znaczący wpływ i w efekcie zmniejszyć skuteczność stada jako formy obronnej przed atakami lub spowolnić proces jego ewolucji.

W eksperymentach 1 - 4, serii III, przeprowadzono badania mające na celu sprawdzenie w jaki sposób różna liczba drapieżników obecnych w środowisku wpływa na przeżywalność oraz tworzenie się zachowań stadnych wśród populacji ofiar. Dodatkowo w badaniach został uwzględniony włączony oraz wyłączony mechanizm komunikacji ofiar.

Do badań wykorzystano wcześniej wyselekcjonowany genotyp drapieżnika, który został wyłączony z procesu dalszej ewolucji. Zamieszczono go w pliku *predator\_rnn\_genes.pkl* (dla sterownika RNN) oraz *predator\_lstm\_genes.pkl* (dla sterownika LSTM), na załączonej do pracy płycie DVD. Warto wspomnieć, że został on wyuczony przy polowaniu na rozproszone ofiary, które nie posiadały umiejętności grupowania się, dlatego uformowanie stada powinno stanowić skutecną formę obrony, nawet w starciu z wieloma drapieżnikami. Symulacja rozpoczyna się od fazy rozgrzewki trwającej 250 kroków.

Wszystkie wartości parametrów środowiska symulacyjnego zawarto w tabeli 4.4. Część z nich, oznaczona jako "parametr dostrajany", jest dostosowywana według potrzeb poszczególnych badań. Parametry środowiska przyjęte na potrzeby eksperymentów z serii III zawiera tabela 5.13. Wartości parametrów algorytmu genetycznego przedstawia tabela 5.11.

Tabela 5.13. Wartości parametrów środowiska dla eksperymentów 1 - 4, serii III

Nazwa parametru	Wartość
<i>Parametry symulacji</i>	
Rozmiar mapy	512 wm x 512 wm
Liczba pokoleń	1000
Liczba kroków symulacji	2000
Początkowa liczba kroków bez drapieżnika	250
<i>Parametry agentów ofiary</i>	
Liczba kopii agenta na początku symulacji	50
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	8
<i>Parametry agentów drapieżnika</i>	
Liczba kopii agenta w symulacji	1 / 2 / 4 / 6
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	8

### 5.3.1. BADANIA DLA STEROWNIKA WYKORZYSTUJĄCEGO SIEĆ RNN

#### Eksperyment 1 - Badanie wpływu liczby drapieżników na przeżywalność ofiar oraz formowanie się wśród nich zachowań stadnych, sterowanych siecią RNN

W eksperymencie wykorzystano miarę średniej przeżywalności ofiar i liczby prób ataków drapieżnika, a także gęstości oraz rozpiętości stada ofiar. Sterownikiem obydwu rodzajów agentów (ofiary i drapieżników) jest sieć RNN. Mechanizm komunikacji ofiar jest wyłączony. Drapieżnicy zostali wyposażeni we wcześniej wyselekcyjowany genotyp oraz wyłączeni z procesu dalszej ewolucji.

#### Wyniki

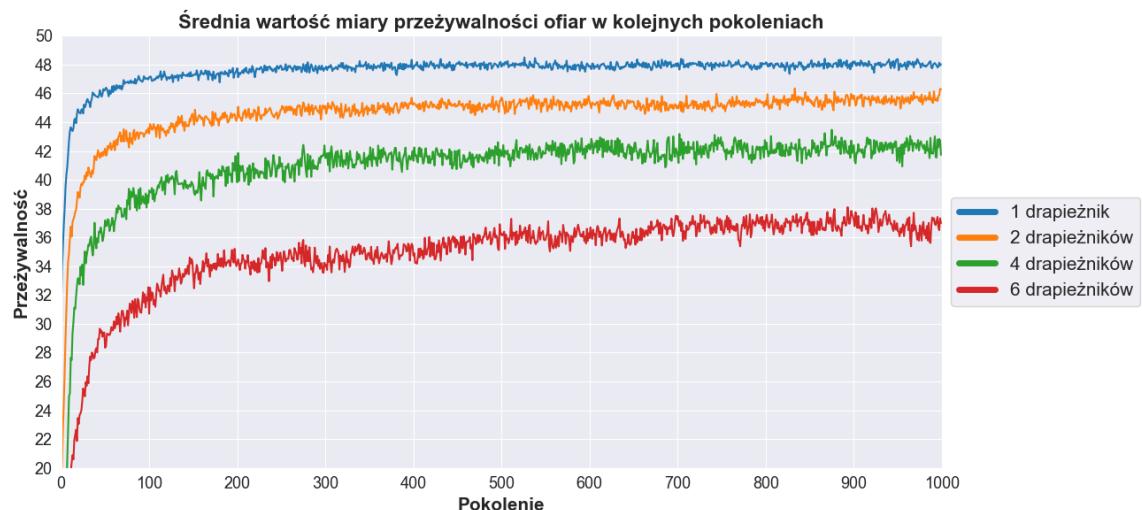
Wyniki eksperymentu zostały zobrazowane na rysunkach 5.44 - 5.47, przedstawiających wykresy średniej wartości wykorzystanych miar w kolejnych pokoleniach. Kolorem niebieskim, pomarańczowym, zielonym, czerwonym i fioletowym, oznaczono wyniki dla środowiska, w którym obecnych było kolejno 1, 2, 4 i 6 drapieżników.

#### Komentarz

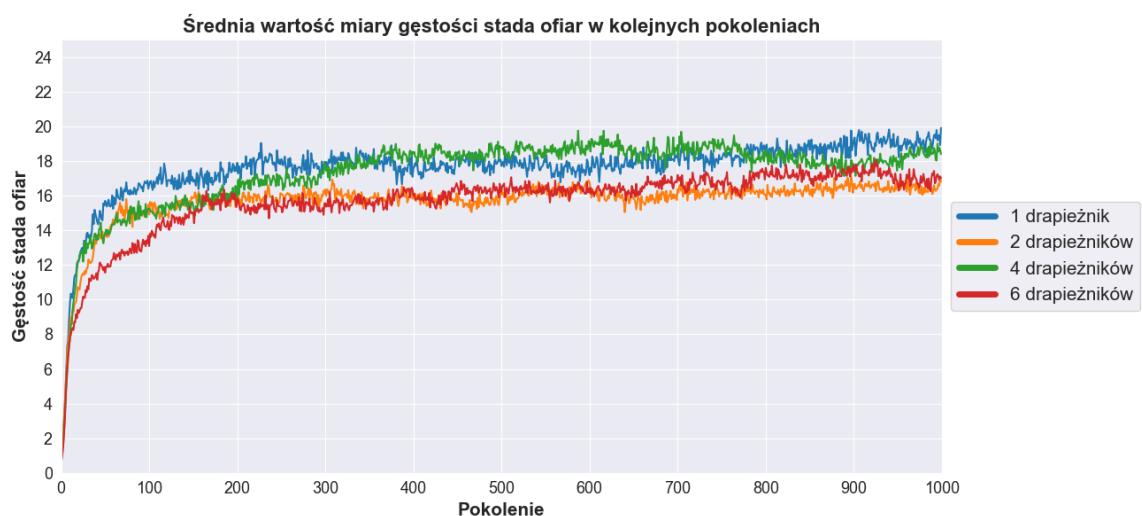
Na podstawie uzyskanych wyników można zauważać, że wraz ze wzrostem liczby drapieżników obecnych w środowisku, proporcjonalnie spada liczba ofiar, którym udało się przetrwać do końca symulacji.

Rysunki 5.45 i 5.46 pokazują, że poszczególne wykresy wartości miar gęstości oraz rozpiętości stada ofiar są do siebie zbliżone. Oznacza to, że stada formowane przez ofiary były podobne, niezależnie od liczby atakujących je drapieżników. Znaczącą różnicę można zauważać na rysunku 5.47, gdzie dla 6 drapieżników średnia liczba prób ataków jest znacznie większa i maleje zauważalnie wolniej w porównaniu do pozostałych przypadków.

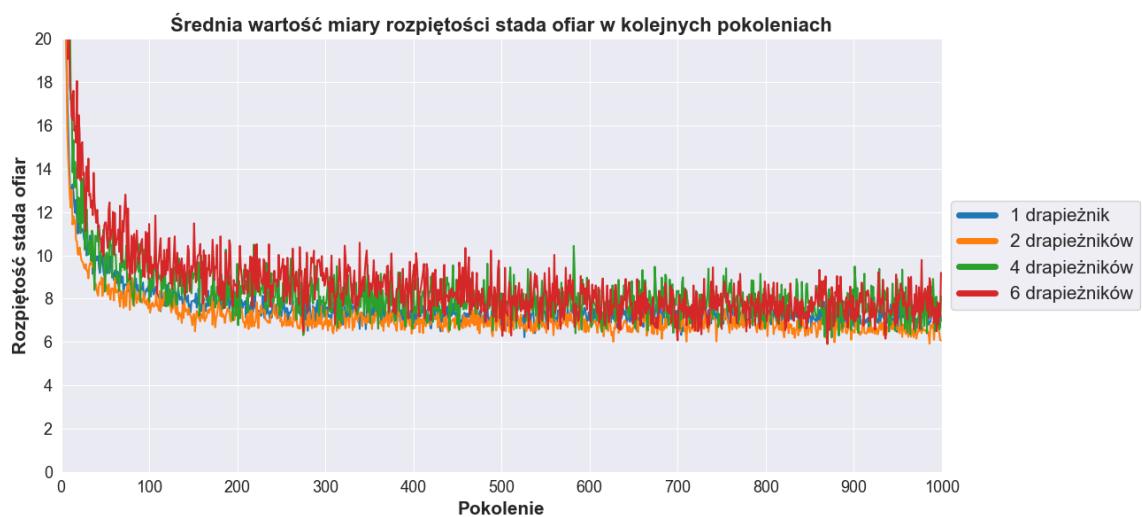
Te obserwacje nie stanowią zaskoczenia i pokazują, że utworzenie stada nie jest ostateczną formą obrony przeciwko drapieżnikom. Mechanizm dezorientacji sprawia, że szansa przeprowadzenia skutecznego ataku na stado ofiar jest niewielka, lecz wciąż prawdopodobna. Większa liczba drapieżników ma proporcjonalnie większe szanse na przeprowadzenie skutecznego ataku, a wraz ze spadkiem gęstości stada, prawdopodobieństwo sukcesu kolejnych ataków wzrasta.



Rys. 5.44. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN.



Rys. 5.45. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN.



Rys. 5.46. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN.



Rys. 5.47. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN.

### Eksperyment 2 - Badanie wpływu liczby drapieżników na przeżywalność ofiar oraz formowanie się wśród nich zachowań stadnych przy włączonym mechanizmie komunikacji ofiar, sterowanych siecią RNN

Do badań zastosowano miarę średniej przeżywalności ofiar, liczby prób ataków drapieżnika oraz gęstości i rozpiętości stada ofiar. Jako sterownik obydwu agentów zastosowano sieć RNN. Komunikacja między ofiarami została włączona. Drapieżniki wyposażono we wcześniej wybrany genotyp oraz wykluczono z procesu dalszej ewolucji.

#### Wyniki

Wyniki eksperymentu przedstawione zostały na rysunkach 5.48 - 5.51, które zawierają wykresy średniej wartości każdej z wykorzystanych miar. W kolorze niebieskim, pomarańczowym, zielonym, czerwonym i fioletowym, przedstawiono wyniki uzyskane dla środowiska, w którym obecnych było kolejno 1, 2, 4 i 6 drapieżników.

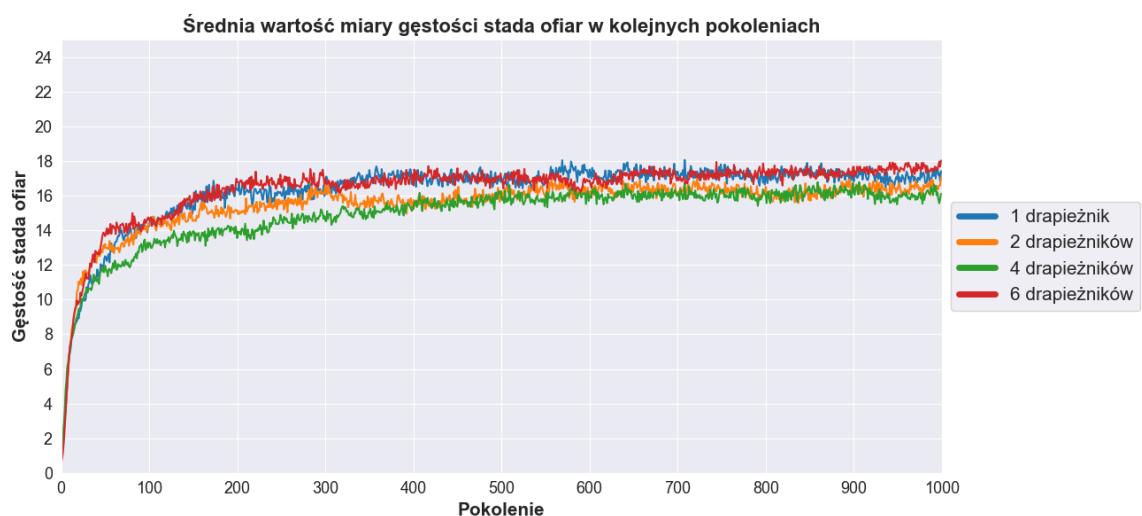
#### Komentarz

Pozitwny wpływ wprowadzenia mechanizmu komunikacji agentów ofiar, wystąpił jedynie w przypadku 6 drapieżników obecnych w środowisku. Średnia przeżywalność ofiar na koniec symulacji, zwiększyła się dzięki niemu aż o 4 osobników. Dla pozostałej liczby drapieżników, miara przeżywalności, gęstości i rozpiętości stada nie uległa znaczącym zmianom.

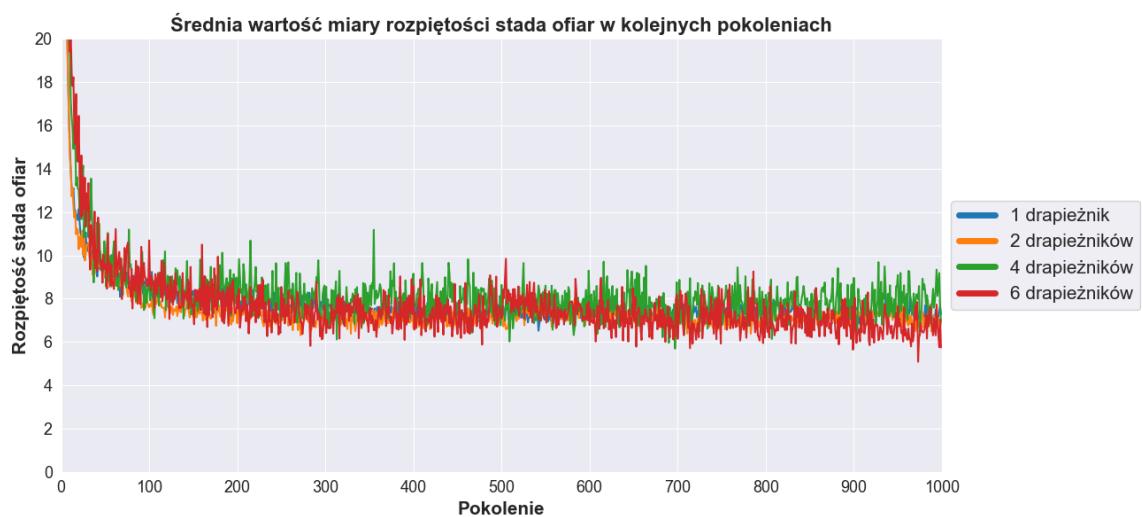
Podobnie jak w poprzednich eksperymentach, komunikacja skutkowała znacznym wzrostem liczby prób ataków drapieżnika. Można również zauważyć, że wykresy miary gęstości oraz rozpiętości stada są do siebie bardziej zbliżone niż w przypadku nieobecnego mechanizmu komunikacji, niezależnie od liczby drapieżników.



Rys. 5.48. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN.



Rys. 5.49. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN.



Rys. 5.50. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN.



Rys. 5.51. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN.

### 5.3.2. BADANIA DLA STEROWNIKA WYKORZYSTUJĄCEGO SIEĆ LSTM

#### Eksperyment 3 - Badanie wpływu liczby drapieżników na przeżywalność ofiar oraz formowanie się wśród nich zachowań stadnych, sterowanych siecią LSTM

Wykorzystano miarę średniej przeżywalności ofiar i liczby prób ataków drapieżnika, a także gęstości oraz rozpiętości stada ofiar. Sterownikiem obydwu agentów jest sieć LSTM. Mechanizm komunikacji ofiar jest wyłączony. Drapieżnikom przypisano wcześniej wyselekcjonowany genotyp. Zostali również wyłączeni z procesu dalszej ewolucji.

#### Wyniki

Wyniki eksperymentu zaprezentowano na rysunkach 5.52 - 5.55, przedstawiających wykresy średniej wartości wykorzystanych miar w kolejnych pokoleniach. Wyniki uzyskane dla środowiska, w którym obecnych było 1, 2, 4 i 6 drapieżników, przedstawiono kolejno w kolorze niebieskim, pomarańczowym, zielonym, czerwonym i fioletowym.

#### Komentarz

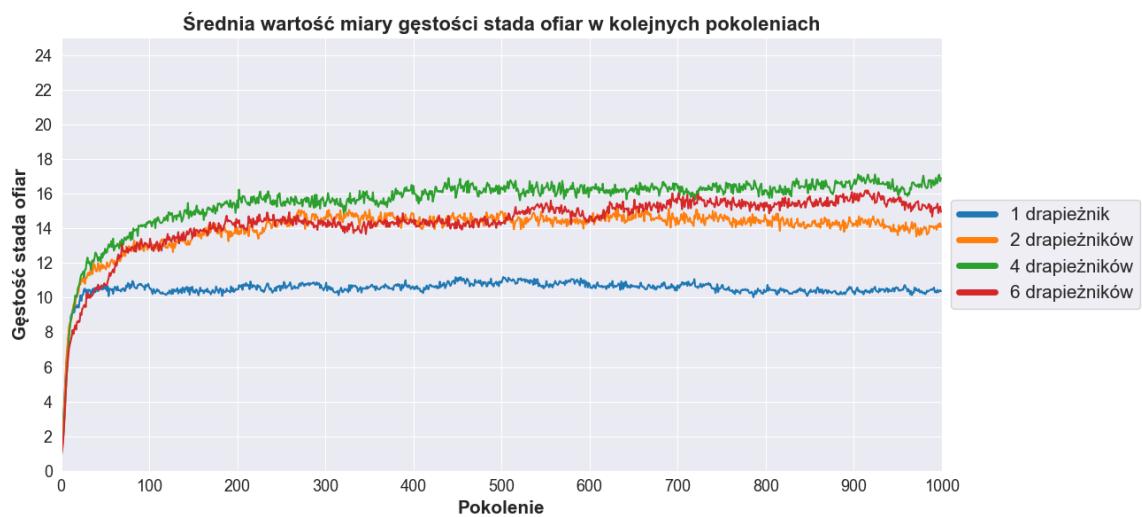
Na podstawie uzyskanych wyników, można zauważyć, że podobnie jak w przypadku sterownika RNN, średnia przeżywalność ofiar maleje wraz ze wzrostem liczby drapieżników, jednak dla 2 i 4 wystąpiła duża zbieżność wykresów.

Porównując wartości miar gęstości oraz rozpiętości stada, tak jak dla sterownika RNN, tworzone przez agentów stada były dość zbliżone. Wyjątek stanowi przypadek z jednym drapieżnikiem, dla którego formowane stada miały najmniejszą gęstość przy jednocześnie najniższej rozpiętości, co sugeruje, że powstających stad było więcej, lecz były mniej liczne. Nie wpłynęło to negatywnie na przeżywalność.

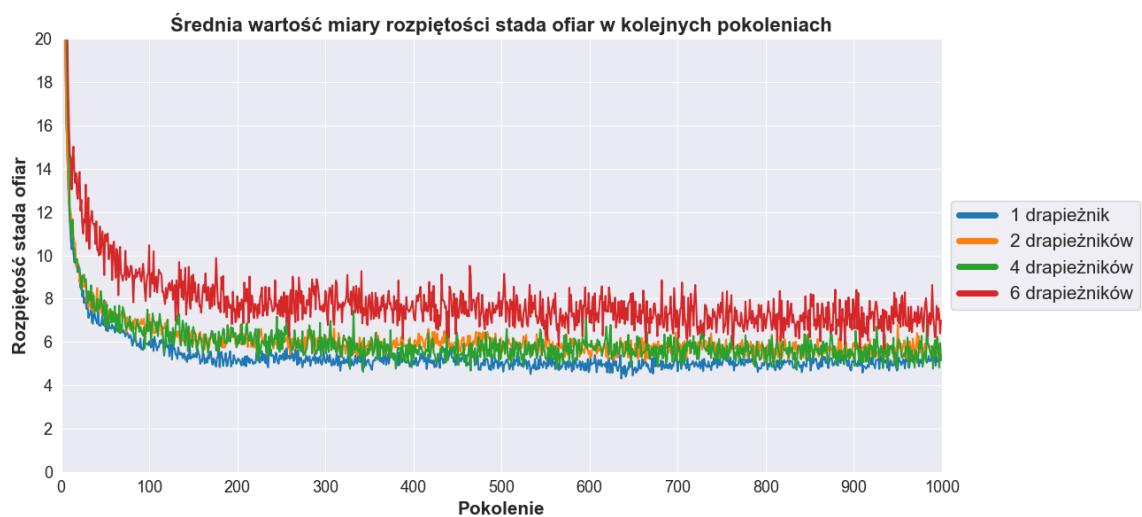
W przypadku miary prób ataków, dla każdej liczby drapieżników obecnych w środowisku, maleje ona wraz z kolejnymi pokoleniami agentów, a końcowe wyniki są dość zbliżone.



Rys. 5.52. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM.



Rys. 5.53. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM.



Rys. 5.54. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM.



Rys. 5.55. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM.

#### **Eksperyment 4 - Badanie wpływu liczby drapieżników na przeżywalność ofiar oraz formowanie się wśród nich zachowań stadnych przy włączonym mechanizmie komunikacji ofiar, sterowanych siecią LSTM**

W eksperymencie wykorzystano miarę średniej przeżywalności ofiar, liczby prób ataków drapieżnika, a także gęstości i rozpiętości stada ofiar. Sterownikiem ofiar oraz drapieżników jest sieć LSTM. Mechanizm komunikacji ofiar został w tym przypadku włączony. Drapieżnicy zostali wyposażeni we wcześniej wyselekcjonowany genotyp oraz wyłączeni z procesu ewolucji.

#### **Wyniki**

Otrzymane wyniki zostały przedstawione na rysunkach 5.56 - 5.59, w formie wykresów średniej wartości wykorzystanych miar w kolejnych pokoleniach. Kolorem niebieskim, pomarańczowym, zielonym, czerwonym i fioletowym, przedstawiono wyniki uzyskane przy obecności kolejno 1, 2, 4 i 6 drapieżników w środowisku.

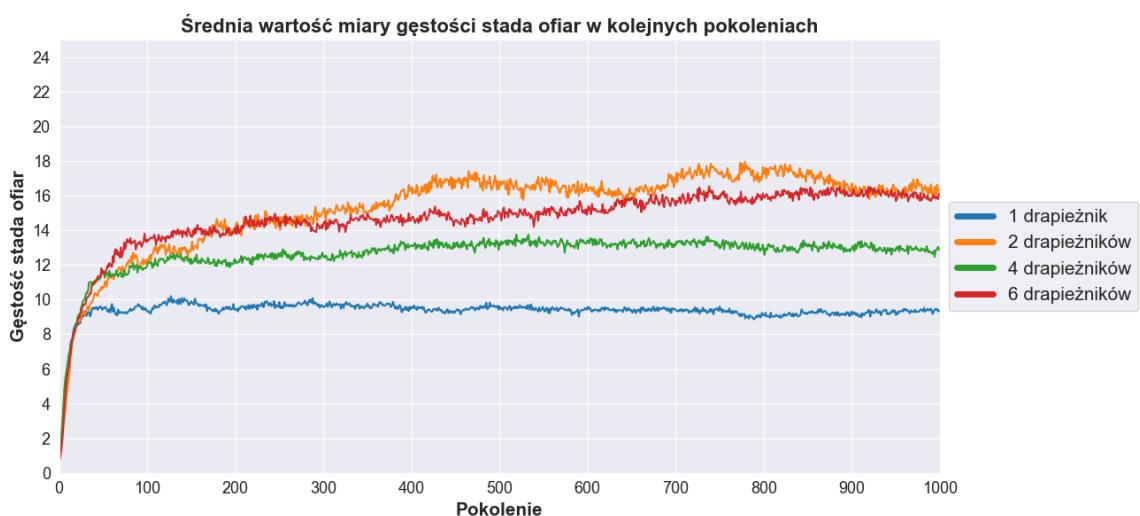
#### **Komentarz**

Niezależnie od liczby drapieżników w środowisku, wprowadzenie mechanizmu komunikacji ofiar, nie spowodowało zmiany średniej przeżywalności. Nie zauważono również wpływu na formowane stada, miary gęstości oraz rozpiętości pozostały bez większych zmian.

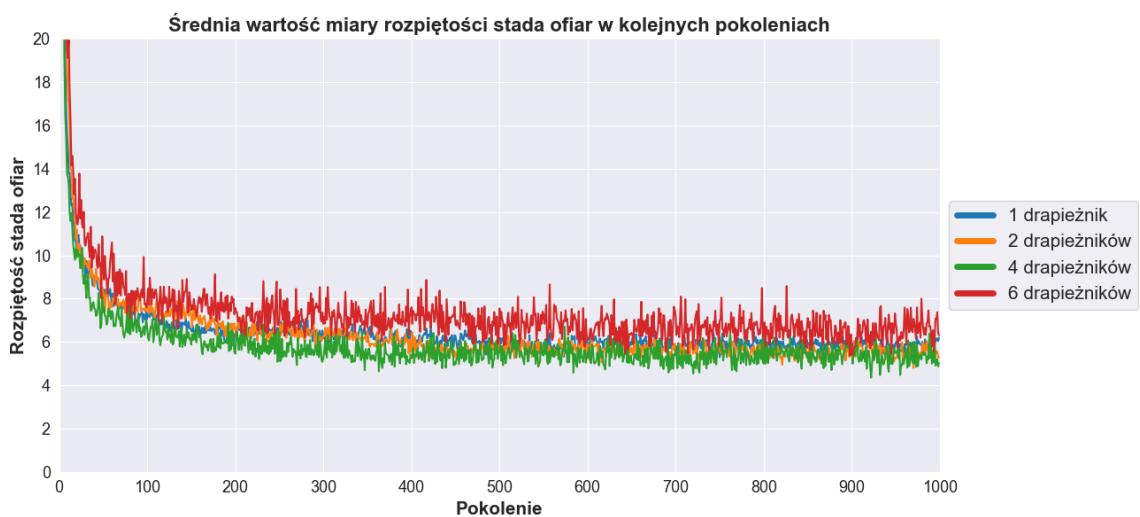
Przy obecnym mechanizmie komunikacji, średnia liczba prób ataków drapieżnika ponownie wzrosła, co pokrywa się z obserwacjami w poprzednich eksperymetach.



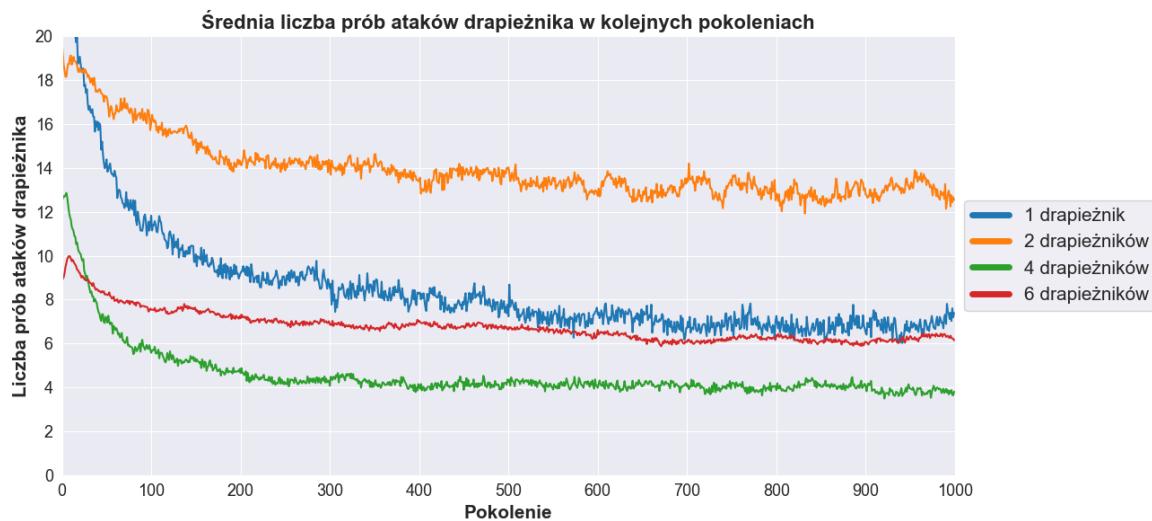
Rys. 5.56. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM.



Rys. 5.57. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM.



Rys. 5.58. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM.



Rys. 5.59. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM.

### 5.3.3. PORÓWNANIE WYNIKÓW EKSPERYMENTÓW SERII III

Wyniki uzyskane w ramach eksperymentów 1 - 4, zostały uśrednione względem liczby drapieżników obecnych w środowisku oraz przedstawione na rysunkach 5.60 - 5.63 w celach porównawczych.

Po uśrednieniu, dla żadnego z zastosowanych sterowników, wprowadzenie mechanizmu komunikacji nie przyniosło pozytywnego wpływu na przeżywalność ofiar, jednak nie zaobserwowano również jej spadku. Jedyną poprawę zaobserwowano dla sterownika RNN przy obecności 6 drapieżników w środowisku.

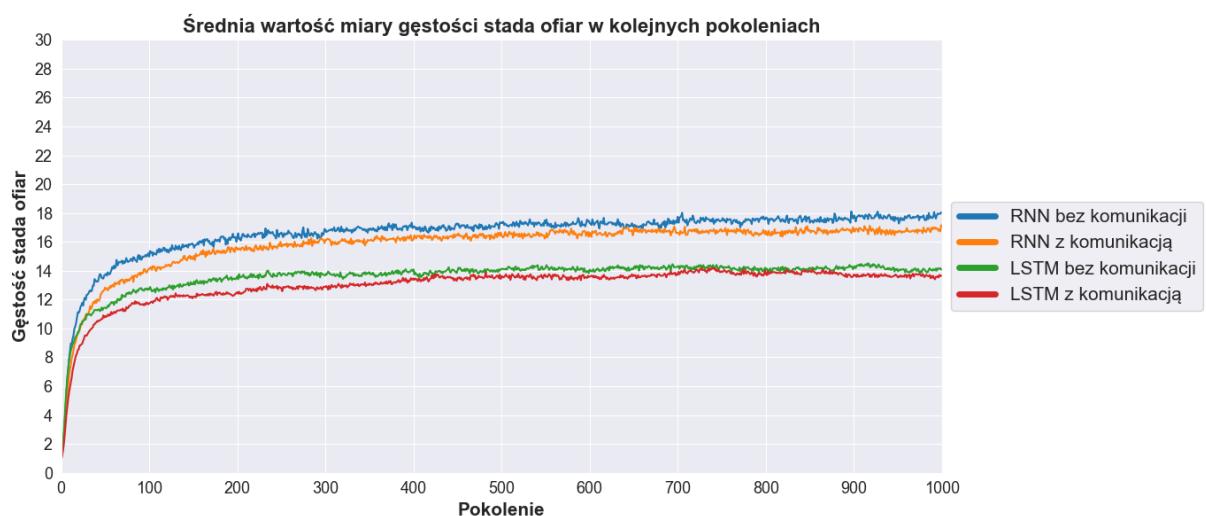
W przypadku sterownika wykorzystującego sieć LSTM, osiągana przeżywalność była wyższa niż dla sterownika RNN. Na tej podstawie nie można jednak stwierdzić, że agenty sterowane siecią LSTM są skuteczniejsze, ponieważ nie jest to bezpośrednie porównanie obydwu sterowników (takie przeprowadzono w VI serii eksperymentów). Ta różnica może wynikać również z tego, że agenci drapieżnika sterowane siecią LSTM mogą sprawować się słabiej.

Na podstawie wykresów gęstości oraz rozpiętości stada (Rys. 5.61 i 5.62), można wyciągnąć podobne wnioski jak w serii II, czyli preferencję agentów sterowanych siecią RNN do formowania gęstszych, lecz mniej ścisłych stad, natomiast w przypadku sieci LSTM, ofiary tworzyły mniej gęste stada i bardziej ścisłe grupy.

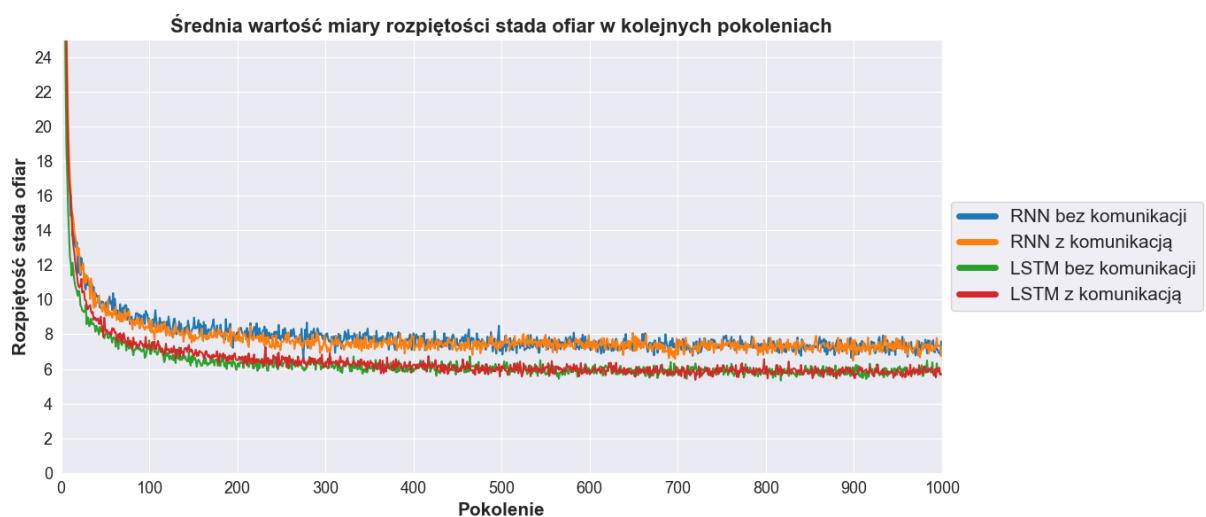
W przypadku średniej liczby prób ataków drapieżnika, sieć LSTM uzyskała zauważalnie lepsze wyniki (mniej ataków), pomimo że na początkowym etapie ewolucji, dla każdego ze sterowników uzyskiwane wartości były podobne. Może z tego wynikać, że agenci sterowane siecią LSTM są w stanie lepiej unikać drapieżników, co tłumaczyłoby również ich wyższą przeżywalność w stosunku do sterownika RNN, jednak tę sytuację tłumaczyłaby również mniejszą skuteczność drapieżników. Te obserwacje zostaną ostatecznie zweryfikowane w serii VI eksperymentów.



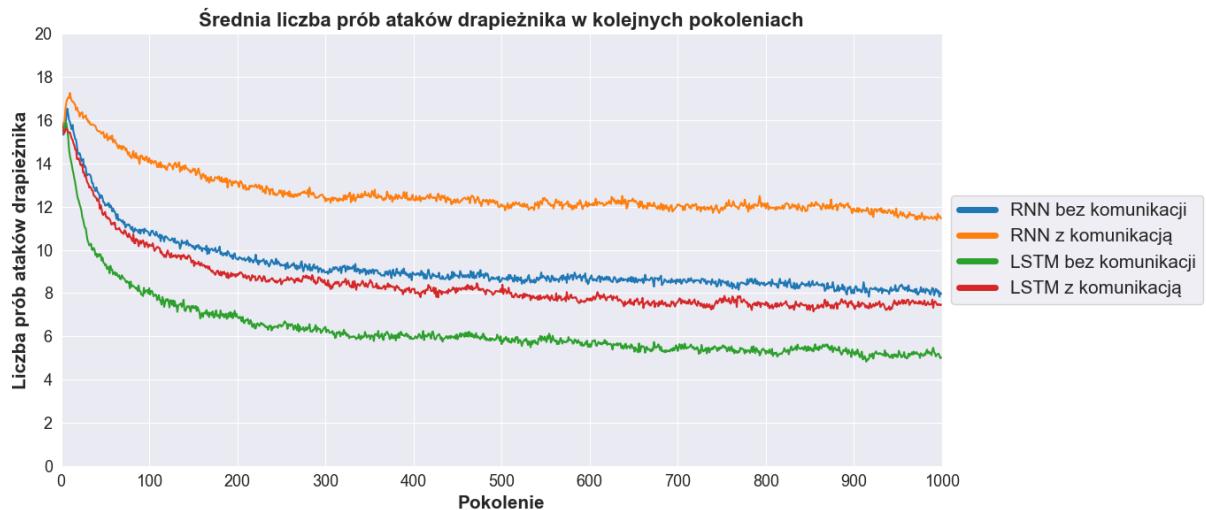
Rys. 5.60. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników.



Rys. 5.61. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników.



Rys. 5.62. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników.



Rys. 5.63. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.  
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników.

#### 5.4. SERIA IV - WPŁYW LICZBY DRAPIEŻNIKÓW OBECNYCH W ŚRODOWISKU NA SKUTECZNOŚĆ PRZEPROWADZANYCH PRZEZ NIE ATAKÓW

Obserwacje prowadzone w naturze pokazują, że wiele gatunków drapieżników poluje w grupach. Ofiary potrafią być od nich szybsze oraz zwinniejsze, a także często żyją w stadach, które zapewniają im przewagę liczebną, dlatego wyłącznie praca zespołowa drapieżników oraz opracowanie odpowiednich strategii, umożliwia przeprowadzenie skutecznego polowania.

W pracach [12, 15], przeprowadzono badania przy obecności wyłącznie jednego drapieżnika, który został wcześniej wyszkolony do polowania na rozproszone ofiary. W takiej sytuacji, ofiary wyewoluowały grupowanie się, jako skuteczną formę obrony przed atakami. Nie przeprowadzono jednak badań, w których drapieżnik miałby możliwość dalszej ewolucji przeciwko grupującym się ofiarom, a tym samym szansę na przezwyciężenie ich obrony. Dodatkowo, większa liczba drapieżników może skutkować wykształceniem strategii wspólnych ataków lub innej formy współpracy, które przyczynią się do zwiększenia skuteczności polowań mimo efektu dezorientacji oraz przełamania formy obronnej stada. Wprowadzenie mechanizmu komunikacji drapieżników również mogłoby im w tym pomóc.

Do przeprowadzenia badań wykorzystano wcześniej wyuczony i wyselekcyjowany genotyp ofiary, w którym wytworzyły się cechy odpowiedzialne za formowanie stada oraz genotyp, w którym ofiary pozostawały rozproszone. Obydwa genotypy, dla każdego ze sterowników można znaleźć na dołączonej do pracy płycie DVD, w plikach *no\_swarm\_prey\_rnn\_genes.pkl* i *swarm\_prey\_rnn\_genes.pkl* (dla sterownika RNN) oraz *no\_swarm\_prey\_lstm\_genes.pkl* i *swarm\_prey\_lstm\_genes.pkl* (dla sterownika LSTM).

Populacja ofiar została wypełniona kopiami wybranego genotypu oraz wykluczona z dalszej ewolucji. Mechanizm komunikacji ofiar nie został uwzględniony. Podobnie jak poprzednio badania poprzedza faza rozgrzewki, dzięki której ofiary mają czas na uformowanie stad. Zostanie sprawdzony wpływ obecności wielu drapieżników w środowisku, a także ich mechanizmu komunikacji na skuteczność ataków w starciu z ofiarami rozproszonymi oraz grupującymi się.

Wszystkie wartości parametrów środowiska symulacyjnego zawarto w tabeli 4.4. Część z nich, oznaczona jako "*parametr dostrajany*", jest dostosowywana według potrzeb poszczególnych badań. Parametry środowiska przyjęte na potrzeby eksperymentów z serii IV zawiera tabela 5.14. Wartości parametrów algorytmu genetycznego przedstawia tabela 5.11.

Tabela 5.14. Wartości parametrów środowiska dla eksperymentów 1 - 4, serii IV

Nazwa parametru	Wartość
<i>Parametry symulacji</i>	
Rozmiar mapy	512 wm x 512 wm
Liczba pokoleń	1000
Liczba kroków symulacji	2000
Początkowa liczba kroków bez drapieżnika	250
<i>Parametry agentów ofiary</i>	
Liczba kopii agenta na początku symulacji	50
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	8
<i>Parametry agentów drapieżnika</i>	
Liczba kopii agenta w symulacji	2 / 4 / 6
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	8

#### 5.4.1. BADANIA DLA STEROWNIKA WYKORZYSTUJĄCEGO SIEĆ RNN

##### Eksperyment 1 - Badanie wpływu różnej liczby drapieżników na skuteczność ich ataków wobec ofiar nie formujących stad, sterowanych siecią RNN

W eksperymencie wykorzystano miarę średniej przeżywalności ofiar i liczby prób ataków oraz rozpiętości drapieżników. Sterownikiem obydwu agentów jest sieć RNN. Agenty ofiar zostały wyposażone w genotyp, który nie powoduje formowania się stad oraz zostały wyłączone z procesu ewolucji. Badaniom poddano drapieżników złączonym oraz wyłączonym mechanizmem komunikacji. Wyjątek stanowi środowisko z jednym drapieżnikiem, ponieważ w jego przypadku mechanizm komunikacji niczego by nie zmienił.

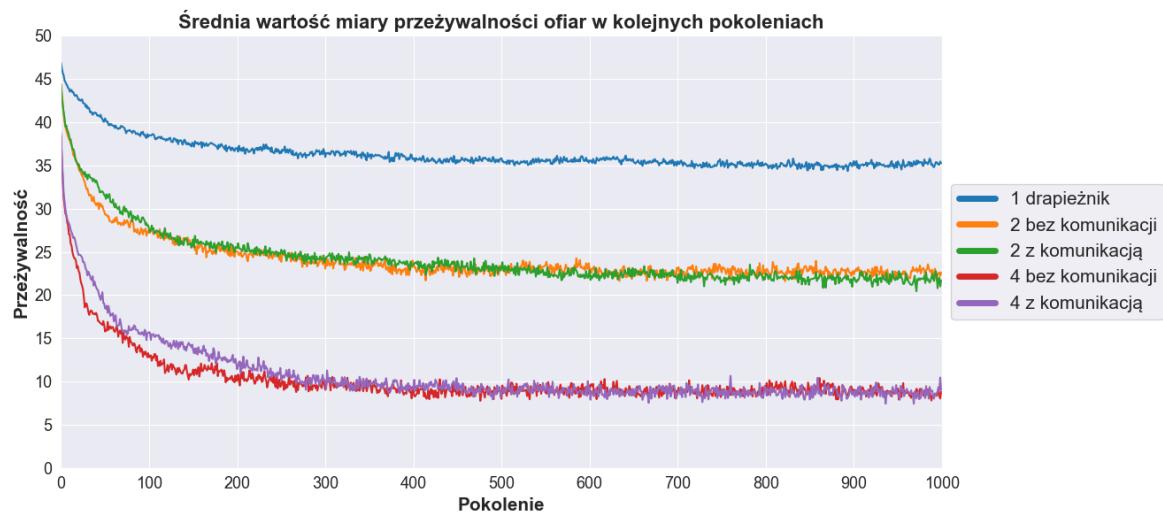
##### Wyniki

Wyniki uzyskane z przeprowadzonego eksperymentu zostały przedstawione na rysunkach 5.64 - 5.66. Zawierają wykresy średnich wartości wykorzystanych miar w kolejnych pokoleniach. Kolorem niebieskim przedstawiono wyniki uzyskane dla jednego drapieżnika, pomarańczowym i czerwonym dla 2 i 4 drapieżników bez mechanizmu komunikacji, natomiast zielonym i fioletowym dla 2 i 4 drapieżników złączonym mechanizmem komunikacji.

##### Komentarz

Uzyskane wyniki pokazują, że wraz ze wzrostem liczby drapieżników, proporcjonalnie malała liczba ofiar, którym udało się przetrwać do końca symulacji. Jest to oczekiwane zachowanie, ponieważ ofiary nie formowały stad i mechanizm dezorientacji nie odgrywał istotnej roli, dlatego drapieżnicy mogli polować swobodnie.

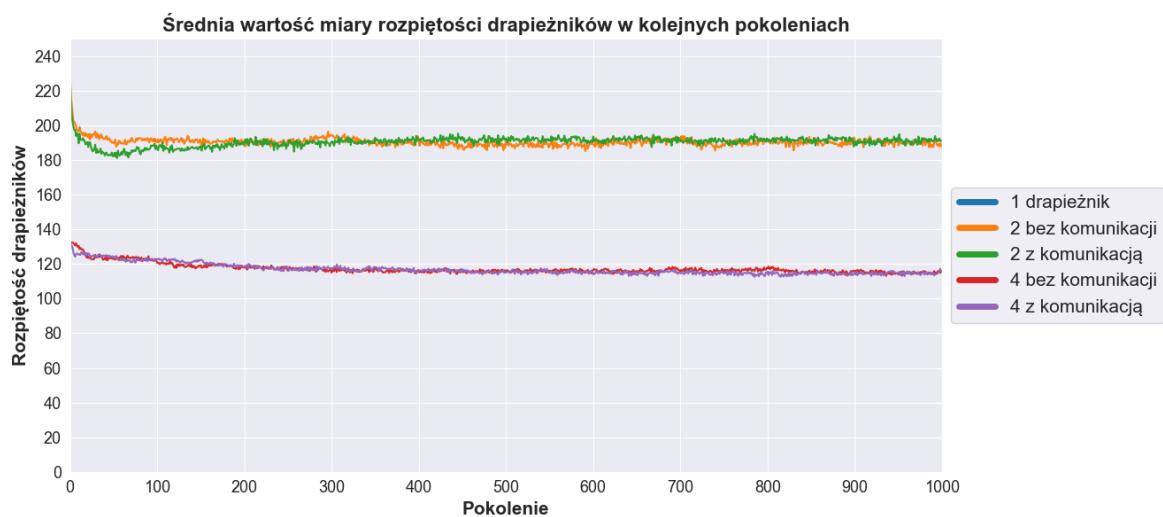
Mechanizm komunikacji nie wpłynął na skuteczność polowania na ofiary, liczbę prób ataków, ani również na średnią minimalną odległość między drapieżnikami (rozpiętość). Wygląda na to, że drapieżnicy nie nauczyli się wykorzystania komunikacji.



Rys. 5.64. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN.



Rys. 5.65. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.  
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN.



Rys. 5.66. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.  
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN.

## **Eksperyment 2 - Badanie wpływu różnej liczby drapieżników na skuteczność ich ataków wobec ofiar formujących stada, sterowanych siecią RNN**

Do badań wykorzystano miarę średniej liczby prób ataków, przeżywalności ofiar oraz rozpiętości drapieżników. Jako sterownik ofiar i drapieżników zastosowano sieć RNN. Agenty ofiar wyposażono we wcześniej wybrany genotyp, umożliwiający formowanie stad. Zostały również wykluczone z dalszej ewolucji. Badaniom poddano drapieżników złączonym oraz wyłączonym mechanizmem komunikacji, za wyjątkiem pojedynczego drapieżnika.

### **Wyniki**

Wyniki badań zobrazowano na rysunkach 5.67 - 5.69, jako wykresy średnich wartości wykorzystanych miar w kolejnych pokoleniach. W kolorze niebieskim przedstawiono wyniki dla pojedynczego drapieżnika, pomarańczowym i czerwonym dla 2 i 4 drapieżników bez mechanizmu komunikacji, natomiast zielonym i fioletowym dla 2 i 4 drapieżników przy obecnej komunikacji.

### **Komentarz**

We wcześniejszych badaniach wykazano, że ofiary były w stanie skutecznie przeciwdziałać atakom drapieżników, którzy przystosowani byli do polowania na ofiary rozproszone. Formacja stada znacząco zmniejszyła skuteczność ataków drapieżnika.

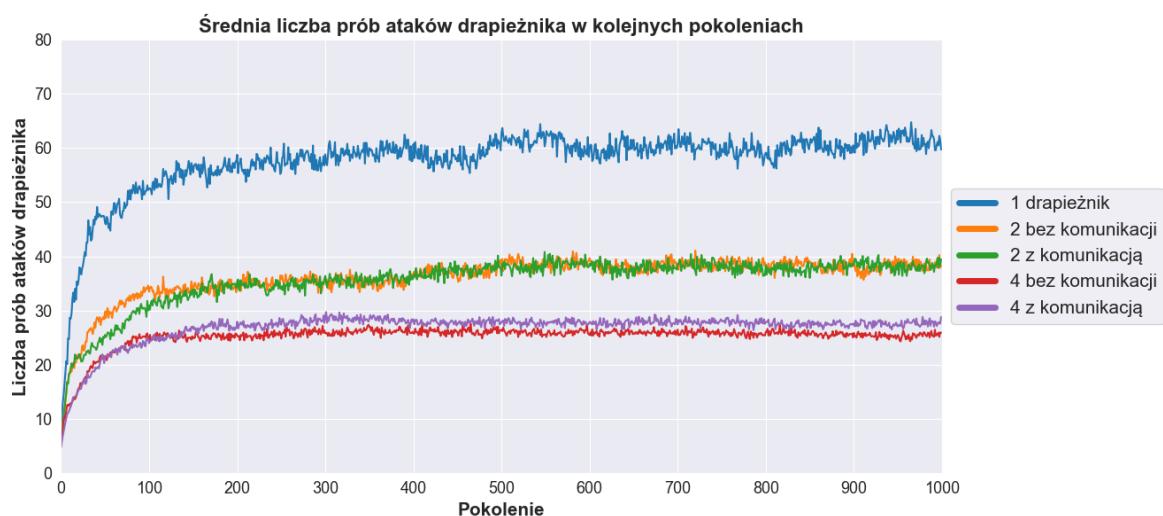
Jednak wyniki przeprowadzone w niniejszym eksperymencie pokazały, że drapieżnicy potrafią wyewoluować w taki sposób, żeby skutecznie polować nawet na grupujące się ofiary. Już pojedynczy drapieżnik był w stanie tego dokonać, a wraz ze wzrostem liczby drapieżników polujących w środowisku, zwiększa się również skuteczność ich ataków, prowadząc do zmniejszenia przeżywalności ofiar jeszcze bardziej. Ponadto, średnia miara przeżywalności ofiar jest podobna, niezależnie od tego, czy ofiary potrafiły formować stada czy też nie (Rys. 5.64 i 5.67).

Na wykresie średniej liczby prób ataków (Rys. 5.68) można zauważyć, że ich liczba znacząco wzrosła, w stosunku do polowania na ofiary rozproszone. W przypadku pojedynczego drapieżnika, zaobserwowany wzrost jest aż trzykrotny. Drapieżnik polujący na stado ofiar, może wykonać wiele ataków w krótkim czasie, bez konieczności poszukiwania nowego celu w środowisku. Ten znaczny wzrost ataków spowodował, że przeżywalność ofiar nie zmalała, pomimo formowania przez nie stad. Warto wspomnieć, że mechanizm dezorientacji jedynie zmniejsza skuteczność ataków przeprowadzanych na grupę ofiar, a nie całkowicie uniemożliwia. W związku z tym, przy znacznym wzroście liczby ataków, nawet jeśli większość z nich okaże się nieskuteczna, drapieżnik wciąż jest w stanie przeprowadzić udany atak. Ponadto, wraz z każdym kolejnym sukcesem, prawdopodobieństwo upolowania dodatkowo wzrasta, ponieważ gęstość stada zmniejsza się.

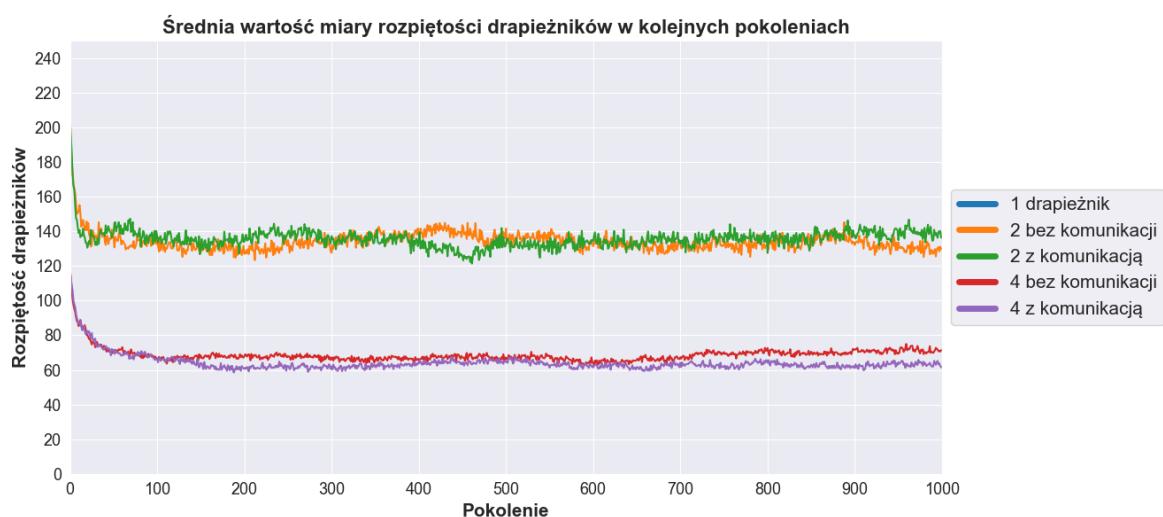
Wizualizacje, utworzone na podstawie symulacji przeprowadzonych w ramach eksperymentu pokazały, że agenci drapieżników polując na grupujące się ofiary, w pierwszej kolejności za cel obierają pojedyncze, oddalające się osobniki lub grupy o mniejszej gęstości. Podczas polowania na zwarte stado, drapieżnicy przeprowadzali bardzo wiele ataków w krótkim czasie. Informacje o dostępności wizualizacji znajdują się w dodatku [7].



Rys. 5.67. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN.



Rys. 5.68. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.  
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN.



Rys. 5.69. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.  
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN.

## **5.4.2. BADANIA DLA STEROWNIKA WYKORZYSTUJĄCEGO SIEĆ LSTM**

### **Eksperyment 3 - Badanie wpływu różnej liczby drapieżników na skuteczność ich ataków wobec ofiar nie formujących stad, sterowanych siecią LSTM**

W eksperymencie zastosowano miarę średniej przeżywalności ofiar, rozpiętości drapieżników oraz liczby prób ataków Sterownikiem agentów jest sieć LSTM. Agenty ofiar wyposażono w genotyp, który nie powoduje formowania się stad. Badaniom poddano drapieżników z włączonym oraz wyłączeniem mechanizmem komunikacji. Wyjątek stanowi środowisko z jednym drapieżnikiem, ponieważ w jego przypadku mechanizm komunikacji niczego by nie zmienił.

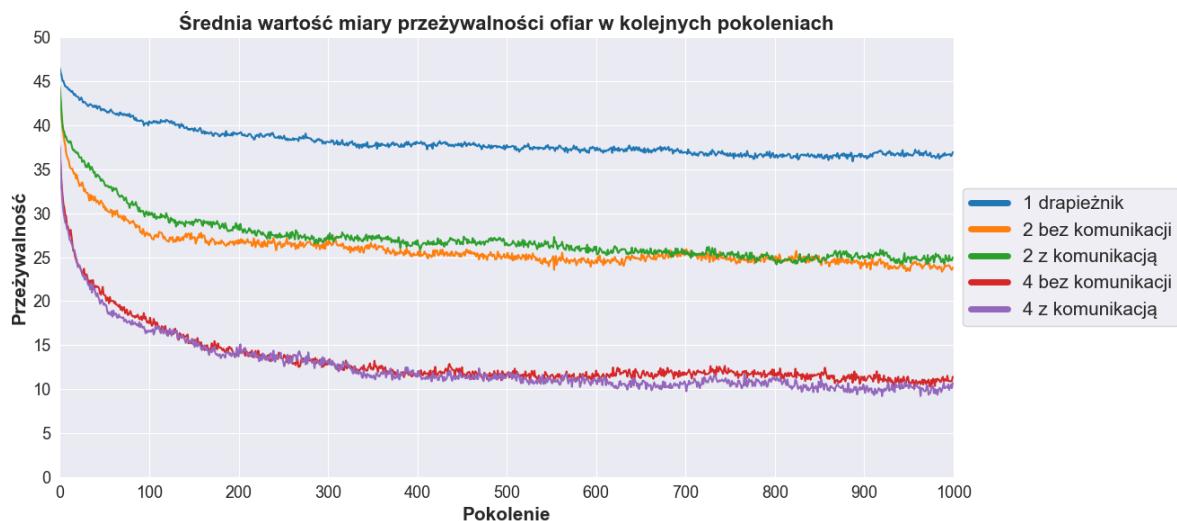
#### **Wyniki**

Uzyskane wyniki zostały przedstawione na rysunkach 5.70 - 5.72, w postaci wykresów średnich wartości wykorzystanych miar w kolejnych pokoleniach. Kolorem niebieskim przedstawiono wyniki uzyskane przy jednym drapieżniku, pomarańczowym i czerwonym dla 2 i 4 drapieżników przy wyłączonej komunikacji, natomiast zielonym i fioletowym dla 2 i 4 drapieżników z obecną komunikacją.

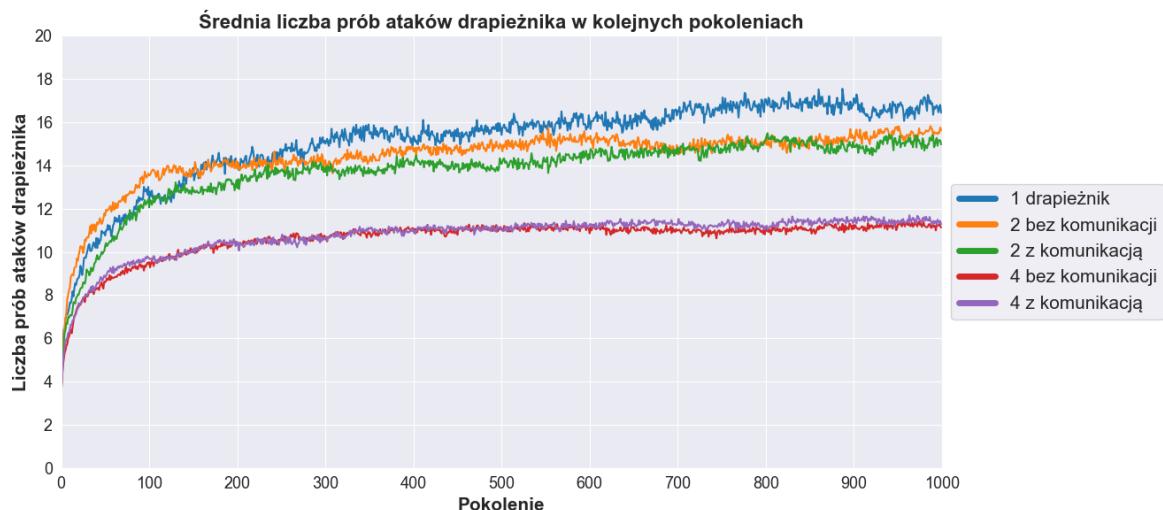
#### **Komentarz**

Na podstawie uzyskanych wyników można stwierdzić, że agenci wyposażone w sterownik bazujący na sieci LSTM, zachowują się bardzo podobnie do agentów sterowanych przez sieć RNN. Analogicznie, drapieżnicy nauczyli się skutecznie atakować stada ofiar. Średnia przeżywalność ofiar maleje wraz ze wzrostem liczby drapieżników obecnych w środowisku.

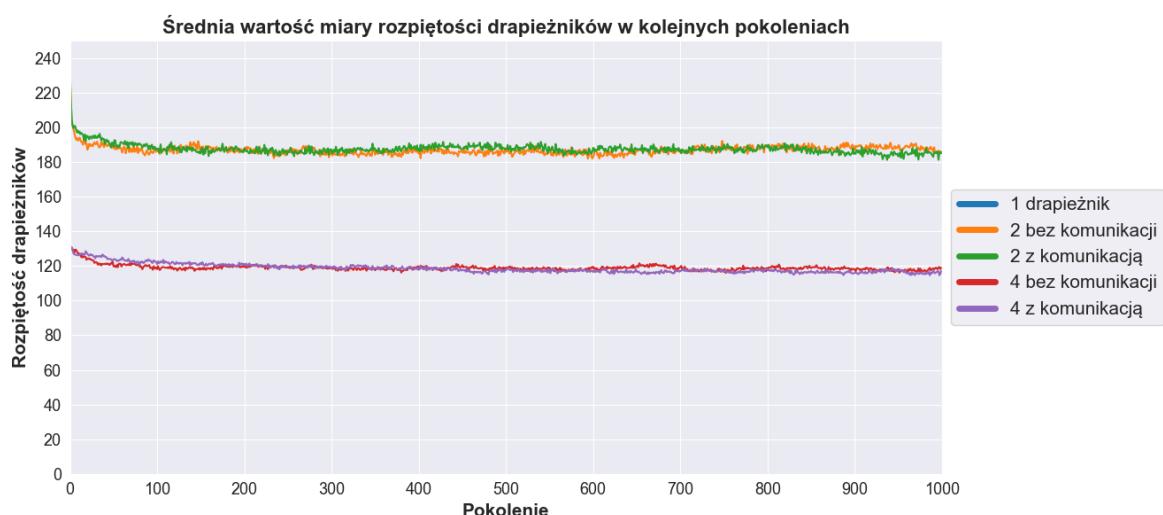
Ponownie nie udało się zaobserwować zmian wynikających z wprowadzenia mechanizmu komunikacji. Najwidoczniej agenci drapieżników sterowane siecią LSTM również nie były w stanie nauczyć się jak go poprawnie wykorzystać lub też nie był on do niczego przydatny.



Rys. 5.70. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach. Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM.



Rys. 5.71. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM.



Rys. 5.72. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach. Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM.

#### Eksperyment 4 - Badanie wpływu różnej liczby drapieżników na skuteczność ich ataków wobec ofiar formujących stada, sterowanych siecią LSTM

W eksperymencie została wykorzystana miara liczby prób ataków drapieżnika, średniej przeżywalności ofiar oraz rozpiętości drapieżników. Sterownikiem obydwu gatunków jest sieć LSTM. Ofiary zostały wyposażone we wcześniej wyselekcjonowany genotyp, w którym wykształciły się cechy odpowiedzialne za formowanie się stad oraz wyłączone z procesu dalszej ewolucji. Badania przeprowadzono dwukrotnie - raz z wyłączonym, a za drugim razem włączonym mechanizmem komunikacji drapieżników.

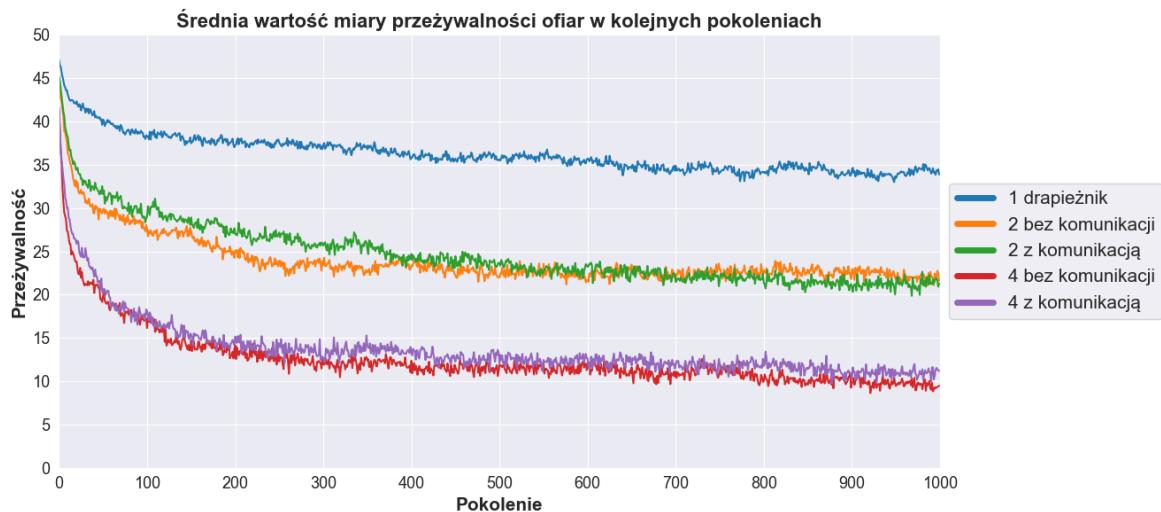
#### Wyniki

Na rysunkach 5.73 - 5.75 zaprezentowano uzyskane wyniki, w formie wykresów średnich wartości wykorzystanych miar dla kolejnych pokoleń. Wyniki uzyskane dla jednego drapieżnika, przedstawiono niebieskim kolorem. Pomarańczowym i czerwonym dla 2 i 4 drapieżników bez mechanizmu komunikacji, a zielonym i fioletowym dla 2 i 4 drapieżników z włączonym mechanizmem komunikacji.

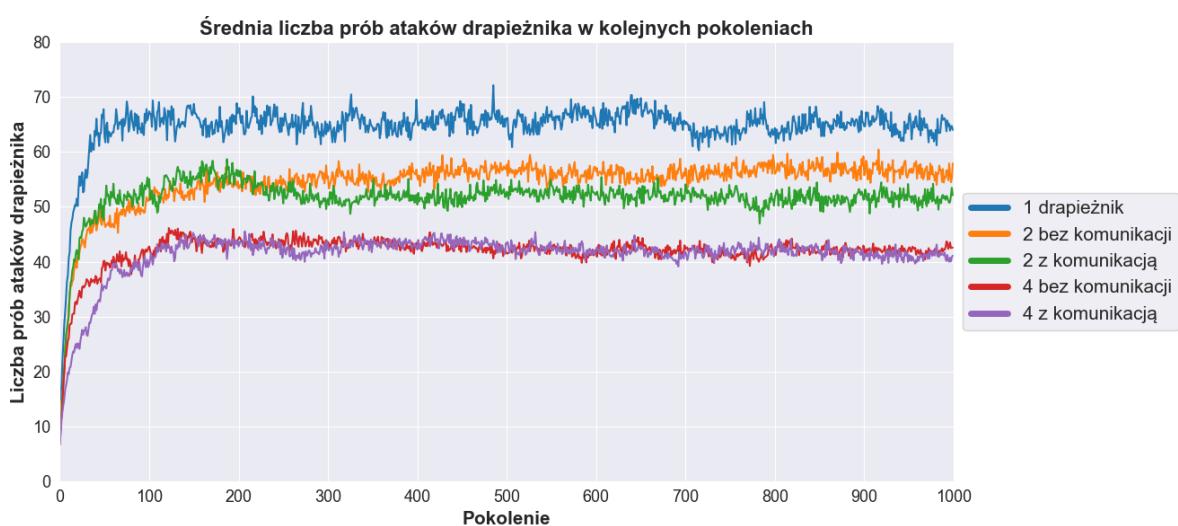
## Komentarz

Otrzymane wyniki są bardzo podobne do tych uzyskanych dla sterownika sieci RNN. Drapieżnicy byli w stanie nauczyć się strategii ataków, dzięki której mogli skuteczniej polować na stada ofiar, pomimo mechanizmu dezorientacji.

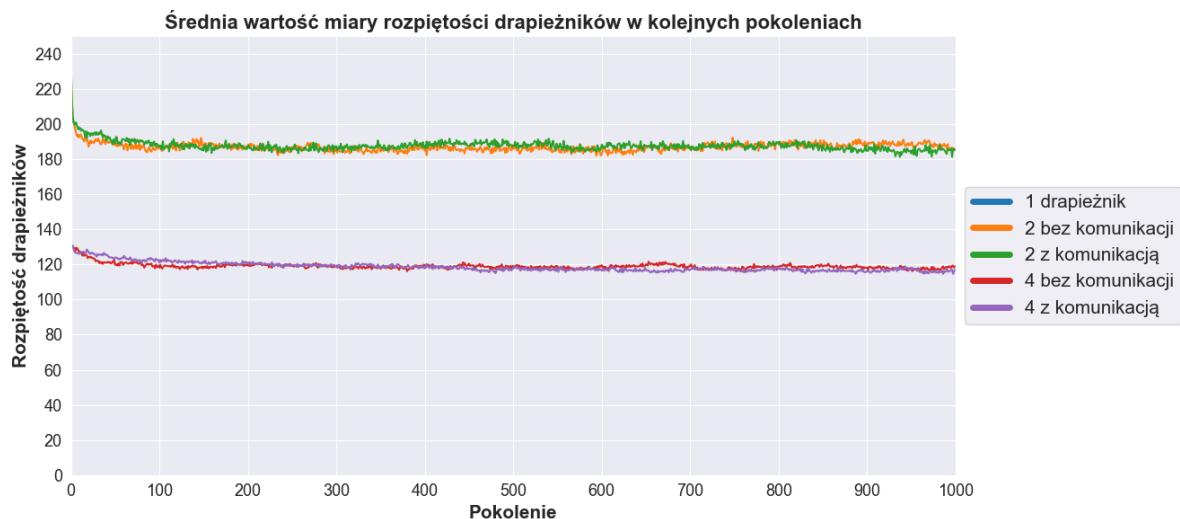
Ponownie nie udało się zaobserwować wpływu komunikacji drapieżników, na wzrost skuteczności polowań czy zmianę w ich zachowaniu (zmniejszenie rozpiętości). Jedyną zauważalną różnicą jest nieznaczny spadek średniej liczby ataków w przypadku dwóch drapieżników, po włączeniu mechanizmu dezorientacji, jednak zmiana ta jest niewielka i może wynikać z przypadku.



Rys. 5.73. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM.



Rys. 5.74. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.  
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM.



Rys. 5.75. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach. Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM.

## 5.5. SERIA V - WPŁYW SPOSOBU ROZMIESZCZENIA POŻYWIENIA W ŚRODOWISKU NA FORMOWANIE SIĘ ZACHOWAŃ STADNYCH U OFIAR

W badaniach prowadzonych w dziedzinie sztucznego życia, najczęściej uwzględniana jest wyłącznie presja drapieżnika. W środowisku naturalnym, każdy organizm musi zadbać o zdobycie pożywienia, do którego dostęp często jest utrudniony, a równocześnie musi zachować czujność i unikać ataku drapieżników, żeby samemu nie stać się pożywieniem dla innych gatunków. Ze względu na to, zwierzęta często decydują się na życie w grupach. Umożliwia to przeszukanie większego obszaru, podnosząc szanse na znalezienie źródła pożywienia, ale również zapewnia ochronę przed drapieżnikami (część osobników może pozostać czujna, podczas gdy inni będą się pożywiać). Niekiedy konieczność zdobycia pożywienia może okazać się silniejsza i osobniki będą zmuszone opuścić bezpieczne schronienie (na przykład stado), w poszukiwaniu nowego żerowiska, narażając się na niebezpieczeństwo.

Badania przeprowadzone w eksperymentach 1 - 6, serii V, mają sprawdzić w jaki sposób rozmieszczenie pożywienia w środowisku oraz obecność drapieżników, wpłyną na zachowanie agentów sterowanych rekurencyjną siecią i czy uda się zaobserwować elementy współpracy ofiar. Dodatkowo zbadany zostanie również mechanizm komunikacji ofiar. Wykorzystano wcześniej wyselekcjonowany genotyp drapieżnika, pochodzący od osobnika, który został wyszkolony do polowania na rozproszone ofiary oraz charakteryzował się najwyższą skutecznością ataków. Można go znaleźć na załączonej do pracy płycie DVD, w pliku *predator\_rnn\_genes.pkl* (dla sterownika RNN) oraz *predator\_lstm\_genes.pkl* (dla sterownika LSTM). Organizmy ofiar zostały wyposażone w energię życiową, która wymusi poszukiwanie pożywienia celem przeżycia. Mechanizm komunikacji drapieżników nie został uwzględniony.

Wzorując się na pracy [24], w badaniach uwzględniono następujące formy rozmieszczenia pożywienia w środowisku:

- Pożywienie rozproszone

Określona liczba pożywienia wynosząca 50, zostaje rozmieszczona po całym obszarze środowiska w sposób zupełnie losowy. Ofiary mają dostęp do pożywienia od początku symulacji. Pożywienie jest uzupełniane na bieżąco, po zjedzeniu nowa roślina zostaje umieszczona w losowym miejscu.

Tabela 5.15. Wartości parametrów środowiska dla eksperymentów 1 - 6, serii V

Nazwa parametru	Wartość
<i>Parametry symulacji</i>	
Rozmiar mapy	512 wm x 512 wm
Liczba pokoleń	1000
Liczba kroków symulacji	2000
Początkowa liczba kroków bez drapieżnika	250
<i>Parametry agentów ofiary</i>	
Liczba kopii agenta na początku symulacji	50
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	8
<i>Parametry agentów drapieżnika</i>	
Liczba kopii agenta w symulacji	2
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	8

- Pożywienie w skupiskach

Losowo zostają wybrane dwa punkty w środowisku, a następnie pożywienie w losowy sposób umieszczane jest w obszarze o promieniu 30wm od nich. W każdym ze skupisk występuje 40 sztuk pożywienia. Ofiary mają dostęp do pożywienia od początku symulacji. Po wyczerpaniu skupiska pożywienia, nowe zostaje wygenerowane w losowym miejscu.

- Pożywienie w pojedynczym skupisku

Agenty początkowo poddawane są rozgrzewce w środowisku ograniczonym do rozmiarów 256wm x 256wm, trwającej 100 kroków. Pożywienie generowane jest w postaci pojedynczego skupiska (80 roślin) o promieniu 30wm, poza granicami mapy. Po etapie rozgrzewki, środowisko jest rozszerzane do domyślnych wartości (512wm x 512wm) i ofiary muszą znaleźć pożywienie, bez wcześniejszej informacji o jego lokalizacji. Po wyczerpaniu źródła pożywienia, nowe ścisłe skupisko roślin jest umieszczane w losowo wybranym miejscu. Energia życiowa agentów została zwiększena do 150, dzięki czemu zyskały dodatkowy czas potrzebny na znalezienie pożywienia.

Wszystkie wartości parametrów środowiska symulacyjnego zawarto w tabeli 4.4. Część z nich, oznaczona jako "*parametr dostrajany*", jest dostosowywana według potrzeb poszczególnych badań. Parametry środowiska przyjęte na potrzeby eksperymentów 1 - 6, serii V, zawiera tabela 5.15. Wartości parametrów algorytmu genetycznego przedstawia tabela 5.11.

### 5.5.1. BADANIA DLA STEROWNIKA WYKORZYSTUJĄCEGO SIEĆ RNN

#### Eksperyment 1 - Badanie wpływu rozproszonego pożywienia na formowanie się zachowań stadnych wśród agentów ofiar przy obecności drapieżników dla sterownika wykorzystującego sieć RNN

W eksperymencie wykorzystano miarę średniej przeżywalności ofiar i liczby zdobytego przez nie pożywienia, a także gęstości oraz rozpiętości stada ofiar. Sterownikiem obydwu rodzajów agentów jest sieć RNN. Drapieżnicy zostali wyposażeni we wcześniej wyselekcjonowany genotyp oraz wyłączeni z procesu dalszej ewolucji. Agenty ofiar posiadają energię życiową malejącą z upływem czasu. Pożywienie występuje w formie rozproszonych po całym środowisku roślin.

## Wyniki

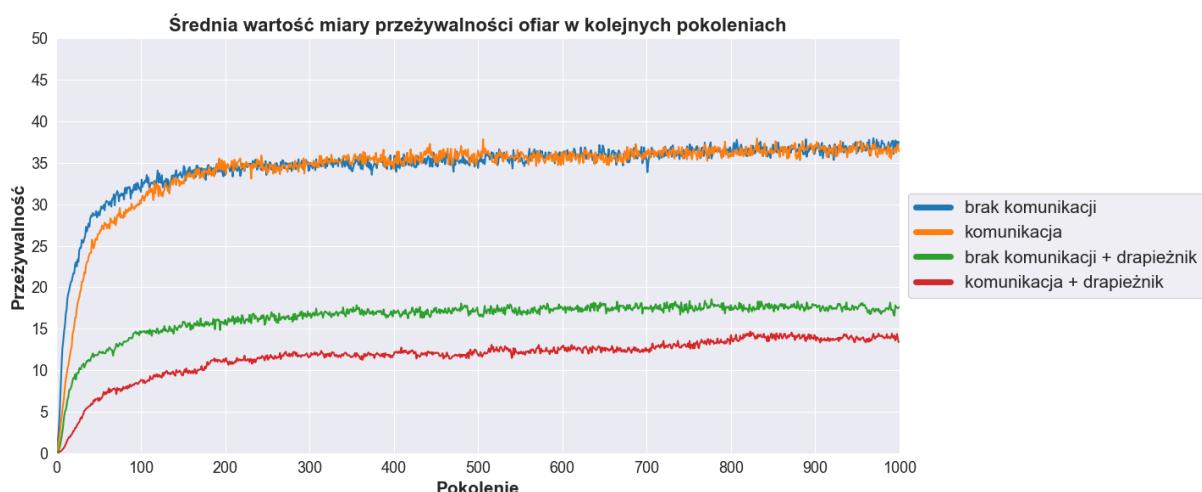
Wyniki uzyskane w ramach eksperymentu przedstawiono na rysunkach 5.76 - 5.79, jako wykresy średnich wartości wykorzystanych miar dla kolejnych pokoleń. W kolorze niebieskim i pomarańczowym przedstawiono rezultaty przy wyłączonym i włączonym mechanizmie komunikacji ofiar, natomiast zielonym i czerwonym odpowiednio przy wyłączonej i włączonej komunikacji oraz obecności drapieżników w środowisku.

## Komentarz

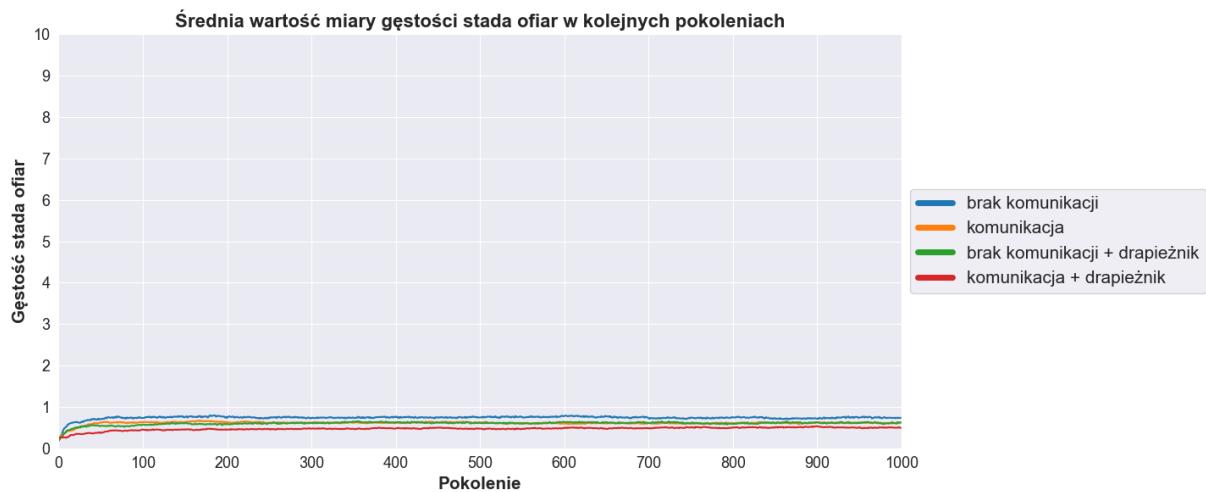
Na podstawie uzyskanych wyników można zauważać, że agenci ofiar poprawnie nauczyły się zdobywać pożywienie, o czym świadczy wzrost jego liczby (Rys. 5.79), a także wzrastająca przeżywalność.

Wprowadzenie do środowiska drapieżników, znacznie zmniejszyło średnią przeżywalność ofiar. Pożywienie występujące w formie rozproszonej stanowiło zbyt silną presję, aby dodatkowo ofiary były zdolne do uformowania stada, przez co były narażone na ciągłe ataki. Można zauważać, że średnia odległość między ofiarami maleje (Rys. 5.78), jednak wciąż są to wartości zbyt wysokie, aby skutecznie przeciwstawić się atakom drapieżników. Mechanizm dezorientacji działa w promieniu 30wm, dlatego zaczyna odgrywać istotną rolę dopiero, kiedy rozpiętość ofiar spadnie poniżej tej wartości, a gęstość stada zacznie wzrastać.

Wprowadzenie mechanizmu komunikacji ofiar nie przyniosło poprawy uzyskiwanych wyników. W środowisku z obecnymi drapieżnikami, został zaobserwowany nawet znaczny spadek przeżywalności. Ofiary nie nauczyły się poprawnie korzystać z dodatkowego zmysłu. Wygląda na to, że dodatkowa informacja w postaci dwóch dźwięków, świadczących o obecności w polu widzenia pożywienia lub drapieżnika, okazała się zbędnym szumem, który wyłącznie przeszkodził w ewolucji agentów i skutkował zmniejszeniem przeżywalności.



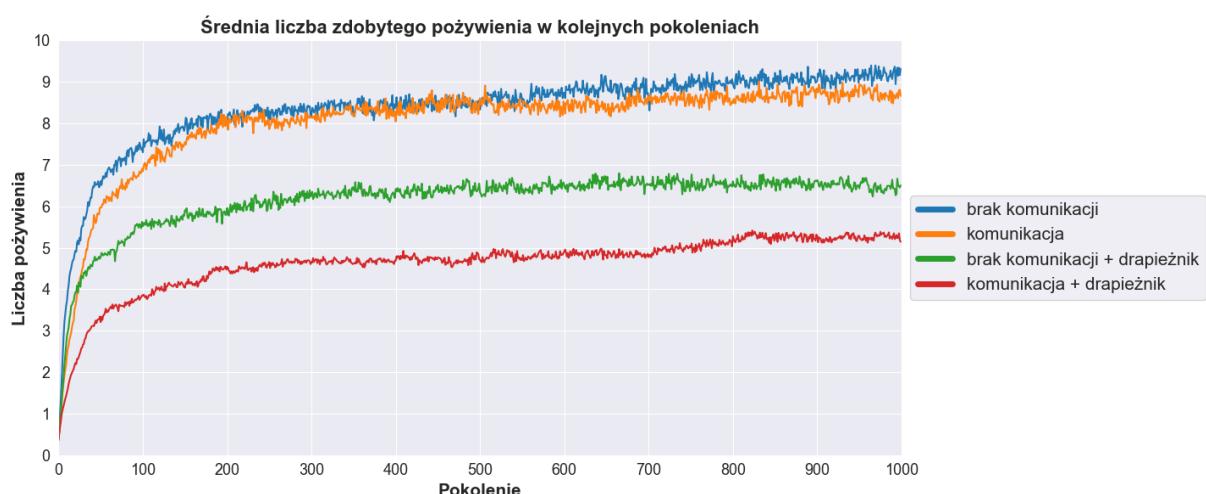
Rys. 5.76. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.



Rys. 5.77. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.



Rys. 5.78. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.



Rys. 5.79. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.

## **Eksperyment 2 - Badanie wpływu pożywienia w postaci skupisk oraz mechanizmu komunikacji na formowanie się zachowań stadnych wśród agentów ofiar przy obecności drapieżników dla sterownika wykorzystującego sieć RNN**

Wykorzystano miarę średniej przeżywalności ofiar i liczby zdobytego pożywienia, a także gęstości i rozpiętości stada ofiar. W roli sterownika obydwu agentów zastosowano sieć LSTM. Drapieżniki wyposażono we wcześniej wybrany genotyp oraz wykluczono z dalszej ewolucji. Ofiary posiadają energię życiową, która maleje w kolejnych krokach symulacji. Pożywienie występuje w formie skupisk roślin.

### **Wyniki**

Wyniki uzyskane w eksperymencie zobrazowano na rysunkach 5.80 - 5.83, jako wykresy średnich wartości wykorzystanych miar. Kolorem niebieskim i pomarańczowym przedstawiono wyniki przy wyłączonym i włączonym mechanizmie komunikacji. Zielonym i czerwonym analogicznie, lecz dodatkowo przy obecności drapieżników w środowisku.

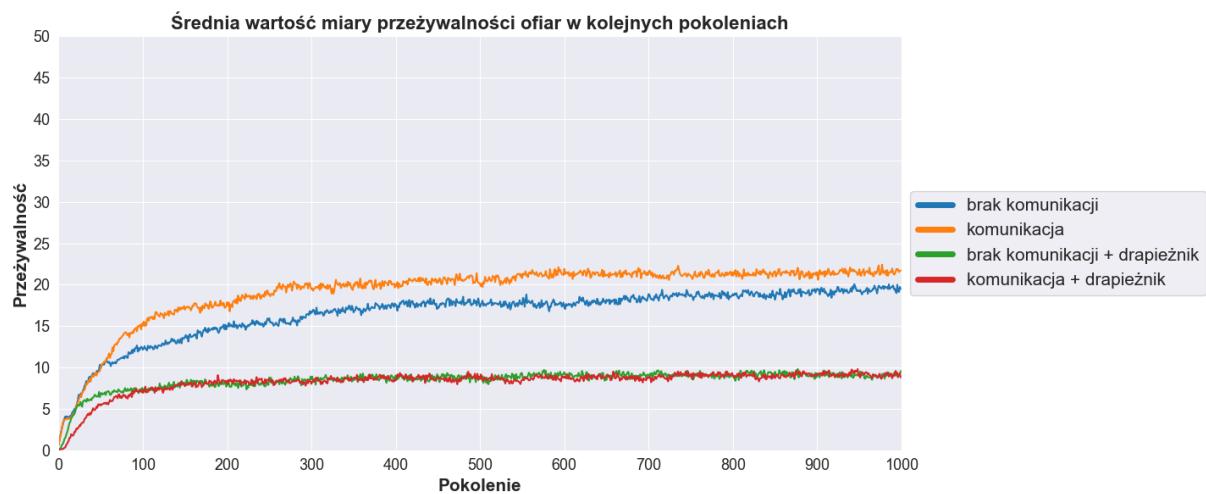
### **Komentarz**

Uzyskane wyniki pokazują, że pożywienie w postaci dwóch skupisk stanowi dla ofiar znacznie większe wyzwanie, co widać po niższych wartościach przeżywalności oraz zdobytego pożywienia. W postaci rozproszonej, pożywienie było praktycznie cały czas w zasięgu wzroku agentów, natomiast w tym przypadku, źródło pożywienia najpierw musi zostać zlokalizowane. Wprowadzenie drapieżników do środowiska jeszcze bardziej utrudniło warunki i sprawiło, że przetrwać były w stanie tylko nieliczne osobniki.

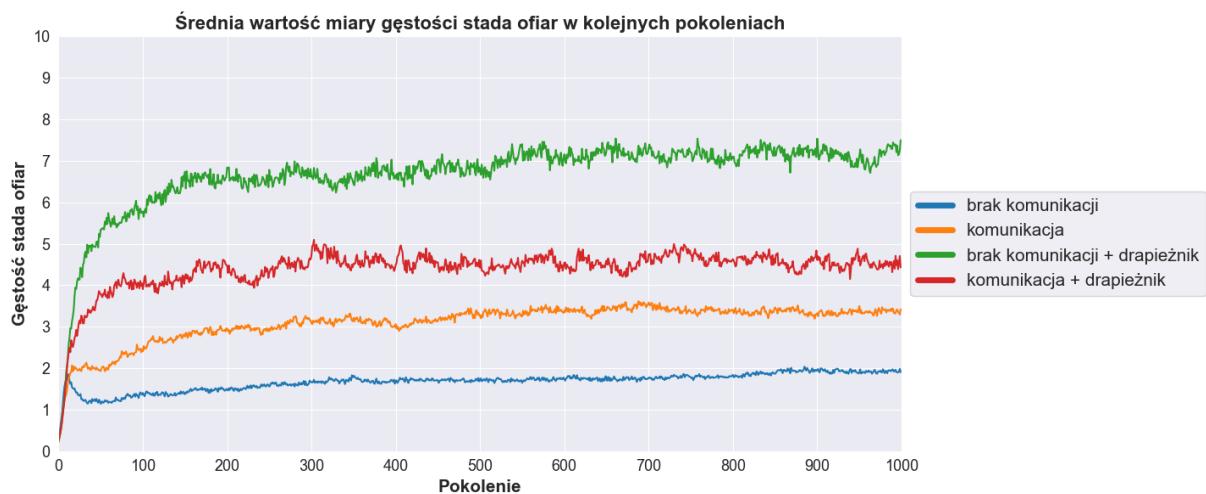
Można jednak zauważyć, że w takich warunkach, wśród agentów ofiar wykształciły się zachowania stadne. Zwłaszcza podczas obecności drapieżników, gęstość stada była dużo wyższa, co świadczy o tym, że ofiary poprawnie zareagowały na ich obecność. Jednakże, nie były w stanie poszukiwać pożywienia oraz chronić się przed atakami drapieżników, przez co ich przeżywalność spadła.

Po raz pierwszy, udało się zaobserwować pozytywny wpływ mechanizmu komunikacji na przeżywalność ofiar, jednak wyłącznie w środowisku bez drapieżników. Wzrosła również gęstość formowanych stad. Ofiary mogły wykorzystać komunikację jako sposób na przyciągnięcie pobliskich osobników do skupiska roślin. W przypadku obecności drapieżników, komunikacja nie wpłynęła na przeżywalność, jednak przyczyniła się do wzrostu liczby zdobywanego pożywienia.

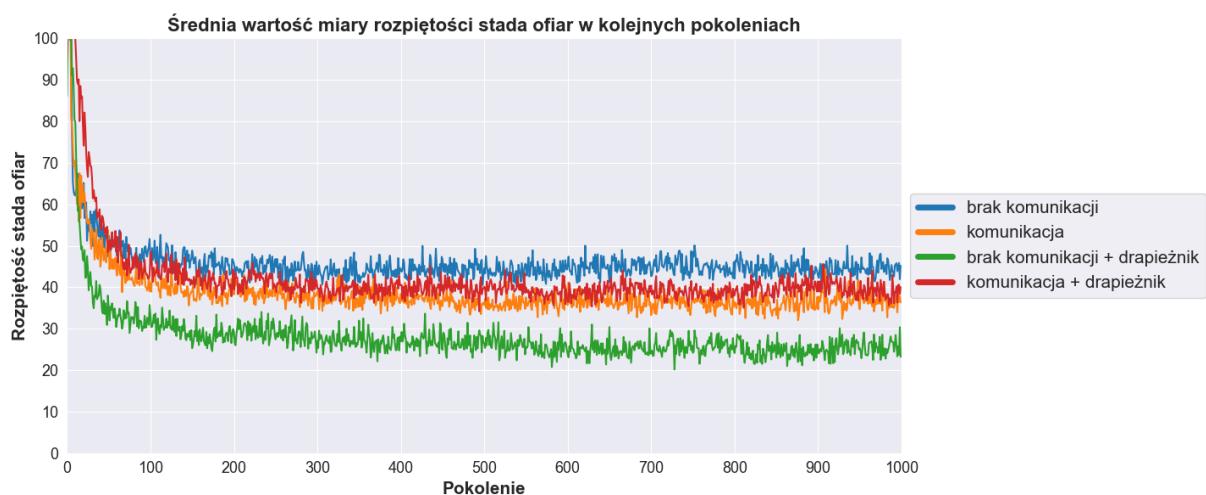
Dalsza analiza z wykorzystaniem wizualizacji pokazała, że agenci istotnie nauczyły się wykorzystywać mechanizm komunikacji w przypadku pożywienia. Zaobserwowano zachowania, w których ofiara zauważała źródło pożywienia, wydała dźwięk oraz wpłynęła na pobliskie osobniki, które po odebraniu sygnału zwracały się w jego kierunku i podążały do źródła pożywienia, wcześniej o nim nie wiedząc. Więcej informacji na temat wizualizacji oraz jej przykłady, można znaleźć w dodatku [7].



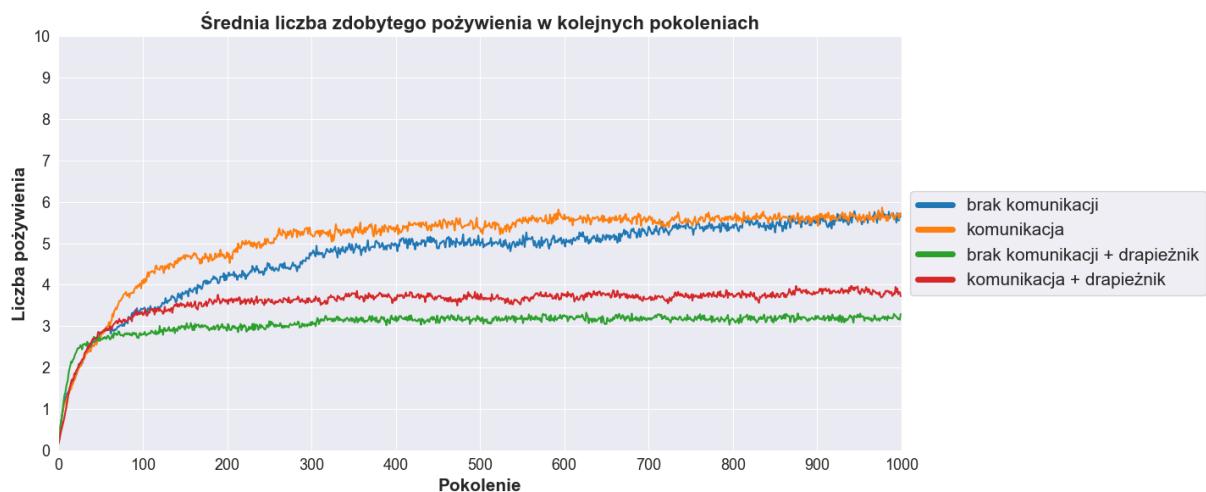
Rys. 5.80. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.



Rys. 5.81. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.



Rys. 5.82. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.



Rys. 5.83. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach. Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.

### Eksperyment 3 - Badanie wpływu pożywienia w postaci pojedynczego skupiska na formowanie się zachowań stadnych wśród agentów ofiar przy obecności drapieżników dla sterownika wykorzystującego sieć RNN

W eksperymencie zastosowano miarę średniej przeżywalności ofiar, liczby zdobytego przez nie pożywienia, a także gęstości i rozpiętości stada ofiar. Jako sterownik agentów wykorzystano sieć RNN. Agenci drapieżników wyposażono we wcześniej wyselekcjonowany genotyp oraz wyłączono z ewolucji. Agenci ofiar posiadają energię życiową malejącą z upływem czasu. Pożywienie dostępne jest w formie pojedynczego skupiska roślin.

### Wyniki

Na rysunkach 5.84 - 5.87 przedstawiono wyniki w formie wykresów średnich wartości wykorzystanych miar w kolejnych pokoleniach. W kolorze niebieskim i pomarańczowym przedstawiono rezultaty przy wyłączonym i włączonym mechanizmie komunikacji ofiar, natomiast zielonym i czerwonym odpowiednio przy wyłączonej i włączonej komunikacji oraz obecności drapieżników w środowisku.

### Komentarz

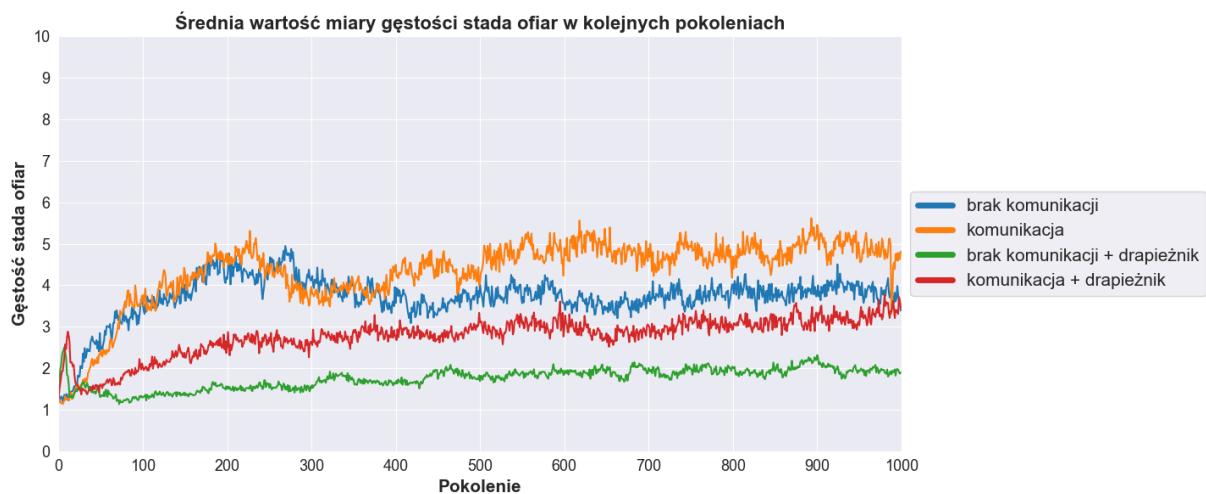
Ta metoda rozmieszczenia pożywienia stanowi dla agentów największe wyzwanie, co można zauważyć na wykresie średniej przeżywalności, która osiąga wartości najmniejsze ze wszystkich trzech metod. Przy obecności drapieżników, przeżywalność spadła jeszcze bardziej.

Sądząc po wykresie rozpiętości oraz gęstości stada, ofiary w procesie ewolucji podjęły próbę tworzenia grup, jednak nie były w stanie nauczyć się jak skutecznie współpracować w celu zlokalizowania pożywienia. Dodatkową trudność sprawiało to, że po wyczerpaniu jednego skupiska roślin, pojawiało się ono w innym, losowym miejscu i ofiary musiały je znaleźć.

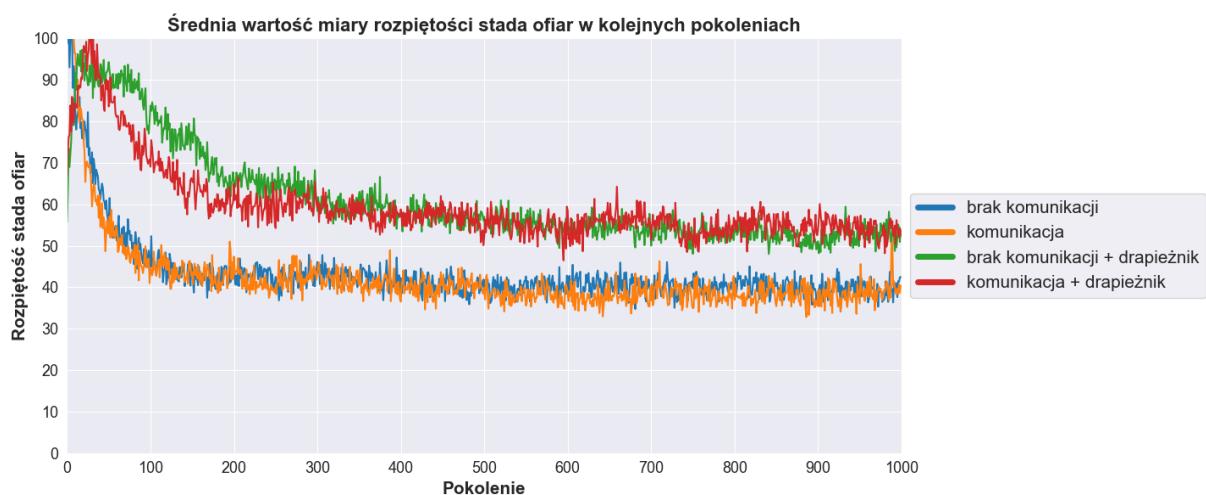
Wprowadzenie mechanizmu komunikacji nieco zwiększyło gęstość stada oraz przyniosło niewielki wzrost przeżywalności, który jednak może być wynikiem przypadku.



Rys. 5.84. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.



Rys. 5.85. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.



Rys. 5.86. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.



Rys. 5.87. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach. Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.

### 5.5.2. BADANIA DLA STEROWNIKA WYKORZYSTUJĄCEGO SIEĆ LSTM

#### Eksperyment 4 - Badanie wpływu rozproszonego pożywienia na formowanie się zachowań stadnych wśród agentów ofiar przy obecności drapieżników dla sterownika wykorzystującego sieć LSTM

Do badań wykorzystano miarę średniej przeżywalności i liczby zdobytego pożywienia przez ofiary oraz gęstości i rozpiętości ich stada. Sterownikiem obydwu agentów jest sieć LSTM. Drapieżniki wyposażono we wcześniej wybrany genotyp. Zostali również wyłączeni z procesu ewolucji. Organizmy ofiar posiadają energię życiową, która maleje wraz z kolejnymi krokami. Pożywienie występuje w formie rozproszych roślin.

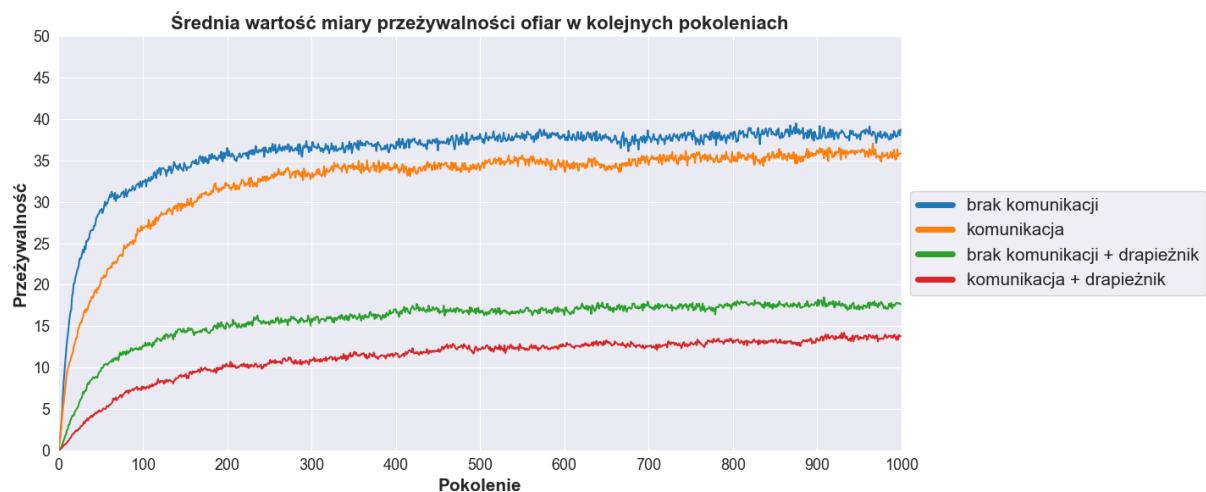
#### Wyniki

Uzyskane w ramach eksperymentu wyniki zostały przedstawione na rysunkach 5.88 - 5.91, jako wykresy średnich wartości wykorzystanych miar. W kolorze niebieskim i pomarańczowym przedstawiono rezultaty przy wyłączonym i włączonym mechanizmie komunikacji ofiar, natomiast zielonym i czerwonym odpowiednio przy wyłączonej i włączonej komunikacji oraz obecności drapieżników w środowisku.

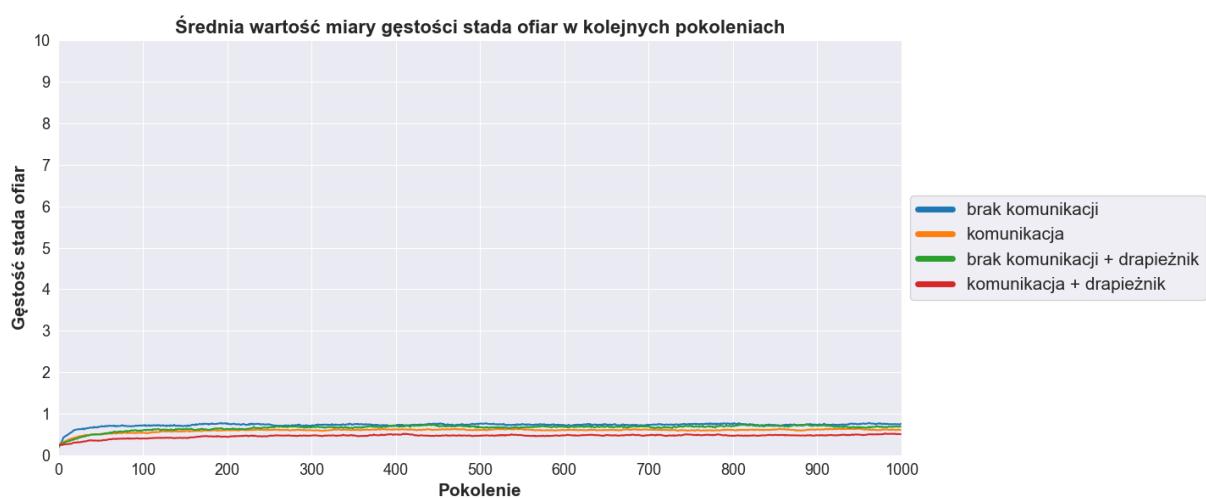
#### Komentarz

Wyniki eksperymentu są podobne do tych uzyskanych dla sterownika RNN. Ofiary poprawnie nauczyły się zdobywać pożywienie, jednak nie były w stanie wyewoluować w kierunku formowania stad, co przełożyło się na duży spadek przeżywalności po wprowadzeniu do środowiska drapieżników.

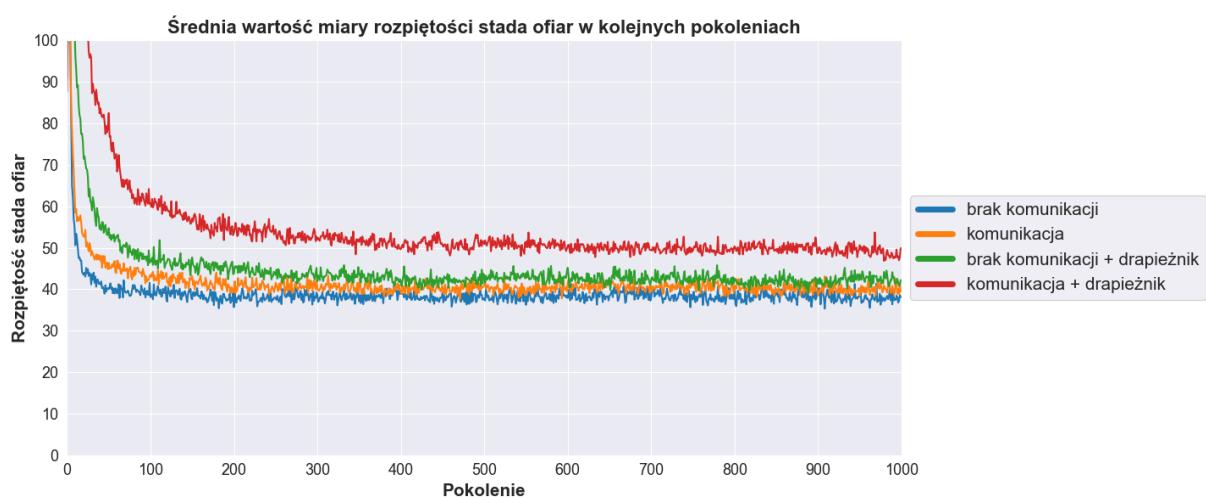
Wprowadzenie mechanizmu komunikacji ponownie nie przyniosło poprawy wyników, lecz wręcz przeciwnie, skutkowało spadkiem średniej przeżywalności ofiar. Ta obserwacja jest niezależna od zastosowanego rodzaju sterownika i wynika ze sposobu rozmieszczenia pożywienia.



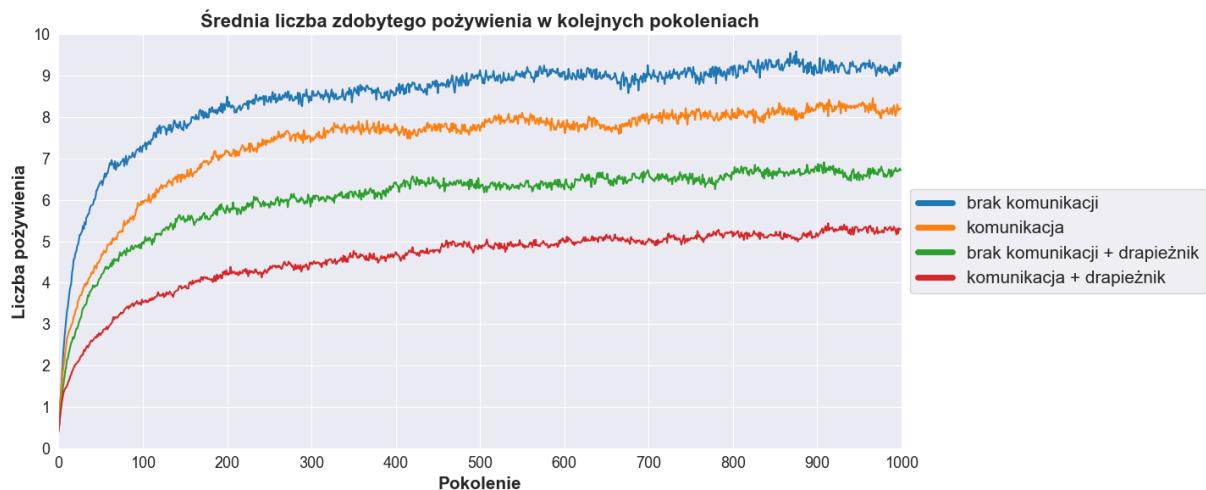
Rys. 5.88. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.



Rys. 5.89. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.



Rys. 5.90. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.



Rys. 5.91. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach. Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.

### Eksperyment 5 - Badanie wpływu pożywienia w postaci skupisk na formowanie się zachowań stadnych wśród agentów ofiar przy obecności drapieżników dla sterownika wykorzystującego sieć LSTM

W eksperymencie wykorzystano miarę średniej przeżywalności ofiar i liczby zdobytego przez nie pożywienia, a także gęstości oraz rozpiętości stada ofiar. Sterownikiem obydwu rodzajów agentów jest sieć LSTM. Drapieżnicy zostali wyposażeni we wcześniej wyselekcjonowany genotyp oraz wyłączeni z procesu dalszej ewolucji. Ofiar posiadają energię życiową malejącą w kolejnych krokach symulacji. Pożywienie występuje w formie skupisk roślin.

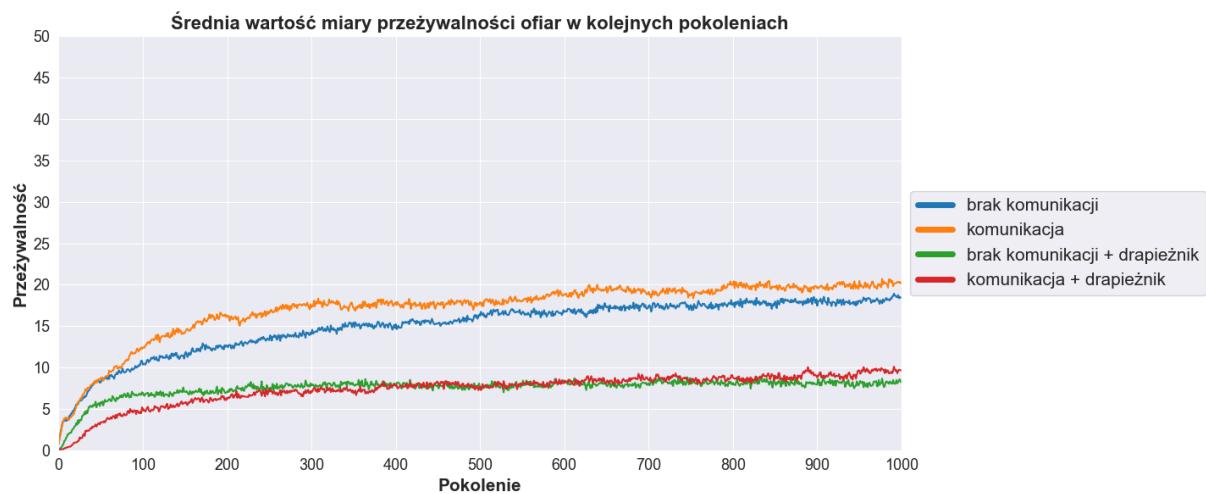
#### Wyniki

Wyniki przedstawiono na rysunkach 5.92 - 5.95, w postaci wykresów średnich wartości wykorzystanych miar w kolejnych pokoleniach. W kolorze niebieskim i pomarańczowym przedstawiono rezultaty kolejno dla wyłączonej i włączonej komunikacji ofiar, natomiast zielonym i czerwonym dla wyłączonej i włączonej, lecz przy obecności drapieżników w środowisku.

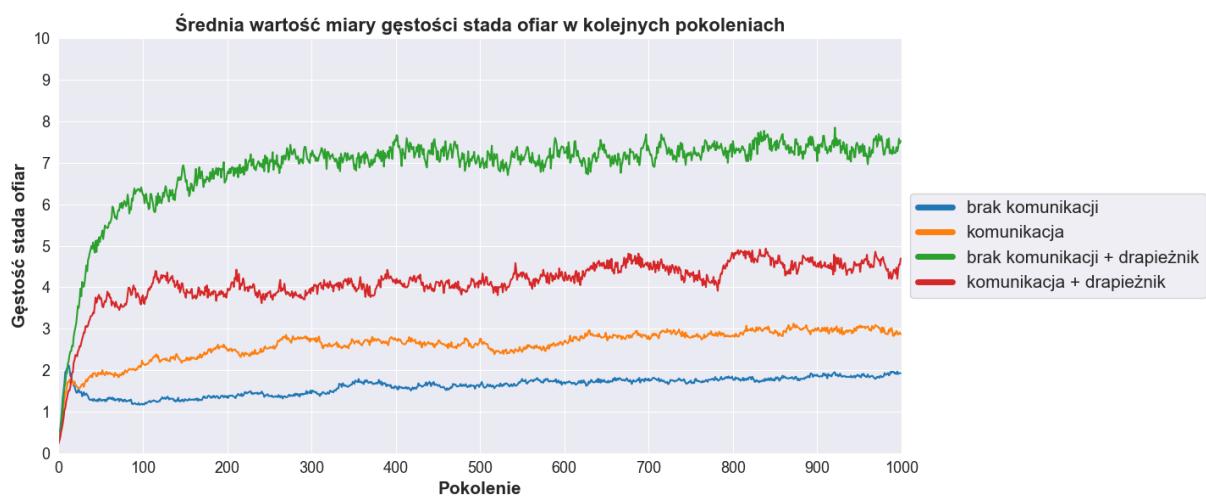
#### Komentarz

W przypadku pożywienia występującego w formie skupisk, uzyskane wyniki również są podobne jak dla sterownika RNN. Takie rozmieszczenie pożywienia stanowiło dla ofiar większe wyzwanie, co skutkowało spadkiem przeżywalności, zwłaszcza podczas obecności w środowisku drapieżników.

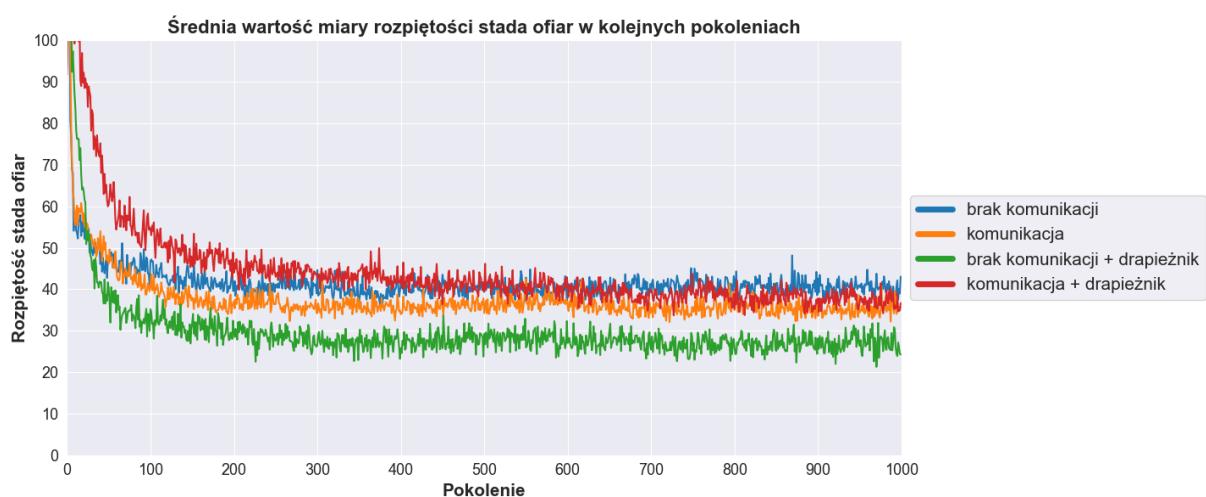
Mechanizm komunikacji ponownie przyczynił się do wzrostu przeżywalności oraz gęstości stada ofiar w środowisku bez drapieżników. Co ciekawe, o ile komunikacja nie wpłynęła na przeżywalność przy obecności drapieżników, to skutkowała prawie dwukrotnym zmniejszeniem miary gęstości stada oraz zwiększeniem średniej odległości między osobnikami (rozpiętości). Może to wynikać z tego, że ofiary nie były w stanie nauczyć się jak zdobyć pożywienie oraz jednocześnie uniknąć ataków drapieżnika, dlatego obrały inną strategię. Po odebraniu sygnału pochodzącego od pobliskiego osobnika, informującego o obecności drapieżnika w pobliżu, próbują się oddalić od jego źródła, żeby uniknąć niebezpieczeństw.



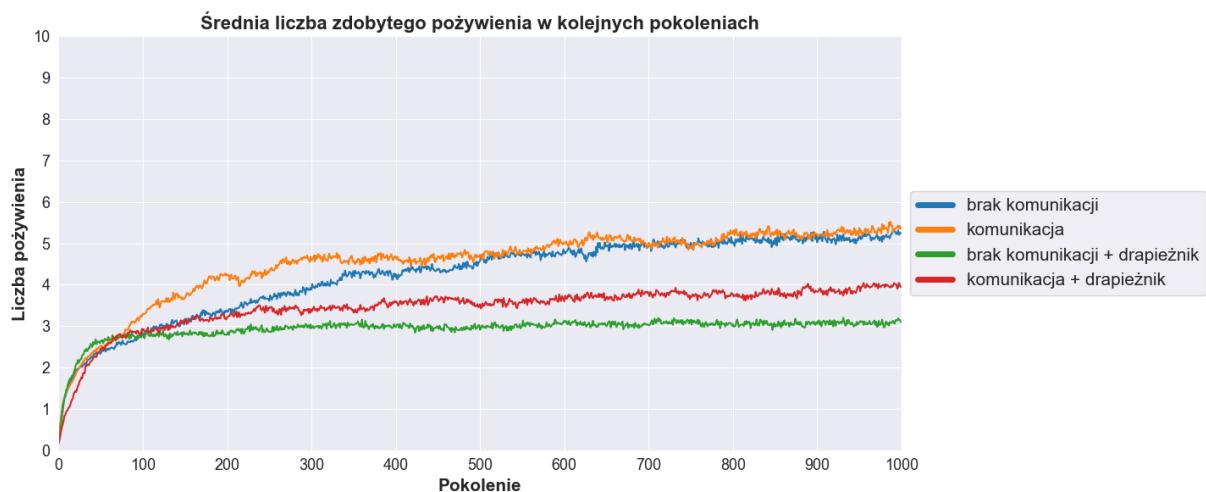
Rys. 5.92. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.



Rys. 5.93. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.



Rys. 5.94. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.



Rys. 5.95. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach. Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.

#### Eksperyment 6 - Badanie wpływu pożywienia w postaci pojedynczego skupiska na formowanie się zachowań stadnych wśród agentów ofiar przy obecności drapieżników dla sterownika wykorzystującego sieć LSTM

W eksperymencie wykorzystano miarę średniej przeżywalności ofiar i liczby zdobytego przez nie pożywienia, a także gęstości oraz rozpiętości stada ofiar. Sterownikiem obydwu rodzajów agentów jest sieć RNN. Drapieżnicy zostali wyposażeni we wcześniej wyselekcjonowany genotyp oraz wyłączeni z procesu dalszej ewolucji. Agenty ofiar posiadają energię życiową malejącą wraz z kolejnymi krokami symulacji. Pożywienie dostępne jest w formie pojedynczego skupiska roślin.

#### Wyniki

Uzyskane wyniki przedstawiono na rysunkach 5.96 - 5.99, w formie wykresów średnich wartości zastosowanych miar dla kolejnych pokoleń. Kolorem niebieskim i pomarańczowym przedstawiono wyniki uzyskane przy wyłączonym i włączonym mechanizmie komunikacji ofiar, a zielonym i czerwonym kolejno przy wyłączonej i włączonej komunikacji, lecz dodatkowo przy obecności drapieżników w środowisku.

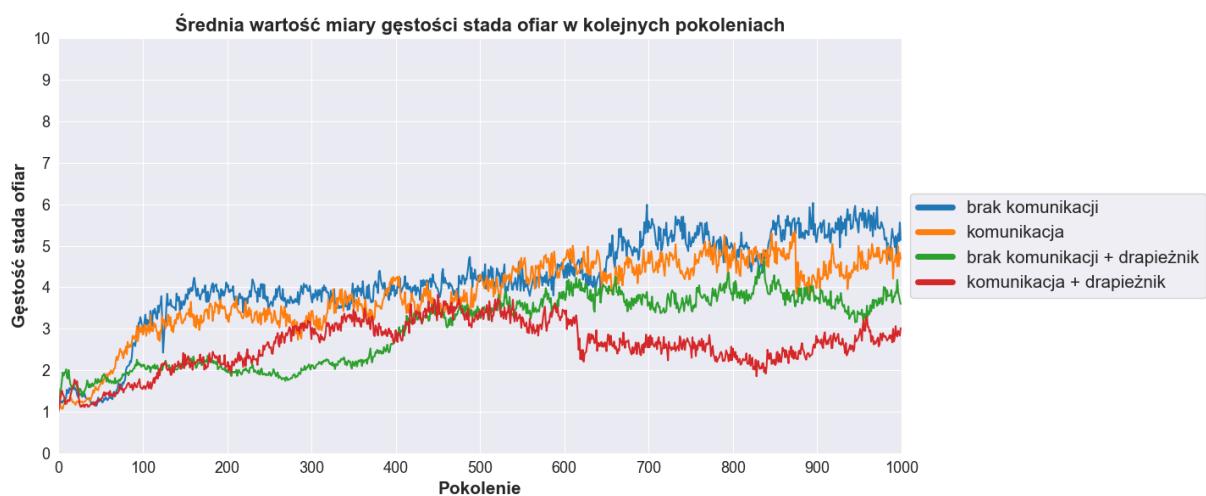
#### Komentarz

Analogicznie jak dla sterownika RNN, przy pożywieniu występującym w postaci pojedynczego skupiska, ofiary nie były w stanie wykształcić zachowań, które pozwoliłyby na skuteczne przetrwanie w środowisku.

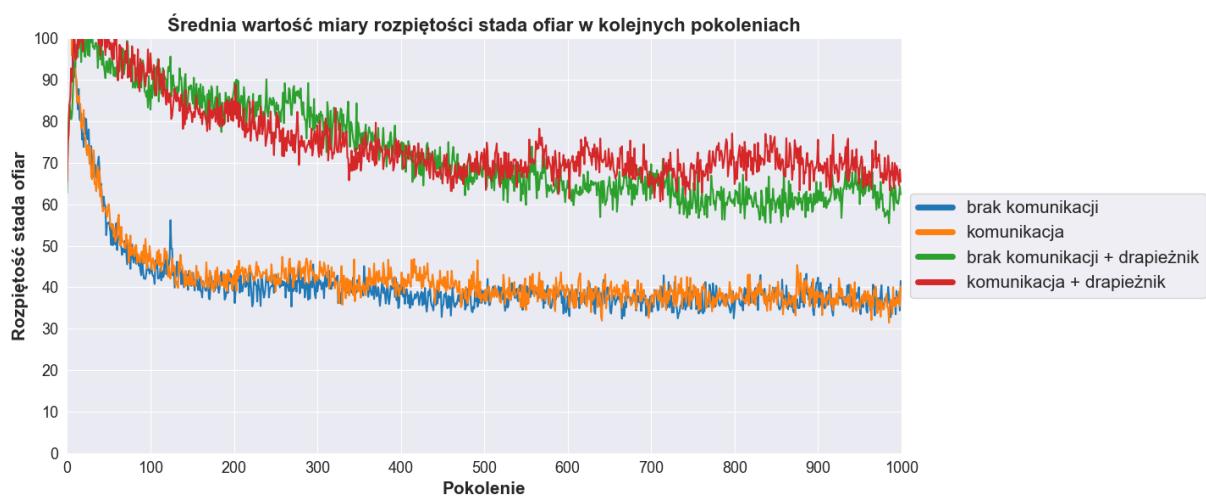
Można zauważyć, że ewolucja podjęła próbę uformowania stad wśród agentów ofiar, o czym świadczy wzrastająca miara gęstości stada oraz malejąca rozpiętość, jednak okazało się to niewystarczające, żeby skutecznie przeciwstawić się atakom drapieżnika czy umożliwić szybsze zlokalizowanie źródła pożywienia. Wprowadzenie mechanizmu komunikacji również nie przyniosło pozytywnego skutku.



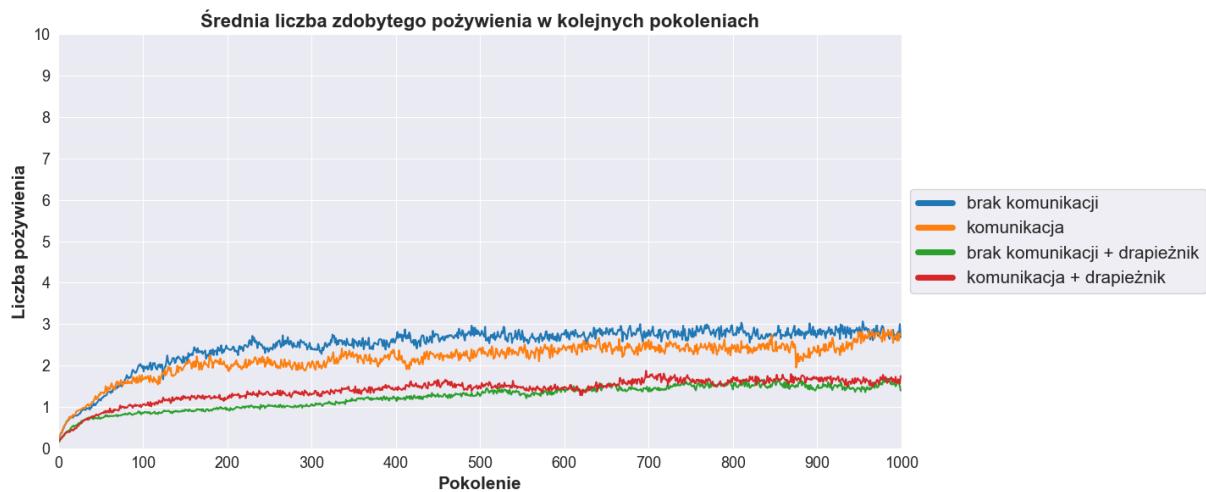
Rys. 5.96. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.



Rys. 5.97. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.



Rys. 5.98. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.



Rys. 5.99. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach. Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.

## 5.6. SERIA VI - BEZPOŚREDNIE PORÓWNANIE STEROWNIKA BAZUJĄCEGO NA SIECI RNN ORAZ LSTM

W ostatniej serii badań, w formie dwóch eksperymentów zostanie przeprowadzone bezpośrednie porównanie obydwu rodzajów sterownika, bazującego na sieci RNN oraz LSTM. W każdym z nich, agenty ofiar oraz drapieżników będą naprzemiennie kontrolowane za pomocą sterownika RNN oraz LSTM.

Pierwszy eksperyment polega na ewolucji ofiar tak, żeby nauczyła się bronić, formować stada w formie obrony przed atakami drapieżników. Zostaną przeprowadzone cztery symulacje. W pierwszych dwóch, ofiary sterowane za pomocą odpowiednio sterownika RNN i LSTM, zmierzą się z drapieżnikiem sterowanym siecią RNN. Kolejne dwa testy przebiegną analogicznie, lecz drapieżnik będzie sterowany siecią LSTM.

W drugim eksperymencie ewolucji zostanie poddany drapieżnik oraz zmierzy się on z ofiarami, które nauczyły się formować stada. W eksperymencie zostaną przeprowadzone ponownie 4 symulacje, analogicznie jak w eksperymencie 1, każdy z agentów będzie naprzemiennie testowany z wykorzystaniem różnego sterownika.

W eksperymetach uwzględniono wyłącznie bez mechanizmu komunikacji, ponieważ wcześniej nie zmieniał on zbyt wiele...

Do przeprowadzenia eksperymentów wykorzystano wcześniej wyselekcjonowany genotyp ofiary, w którym wytworzyły się cechy odpowiedzialne za formowanie stada oraz genotyp drapieżnika, który wyuczony został do polowania na rozproszone ofiary. Zależnie od badania, jedna z populacji agentów (ofiary lub drapieżników), została wyposażona w wybrany genotyp, odpowiedni dla rodzaju zastosowanego sterownika oraz wyłączona z dalszej ewolucji. Można je znaleźć w plikach *swarm\_prey\_rnn\_genes.pkl* i *predator\_rnn\_genes.pkl* (dla sterownika RNN) oraz *swarm\_prey\_lstm\_genes.pkl* i *predator\_lstm\_genes.pkl*, na załączonej do pracy płycie DVD.

Wszystkie wartości parametrów środowiska symulacyjnego zawarto w tabeli 4.4. Część z nich, oznaczona jako "parametr dostrajany", jest dostosowywana według potrzeb poszczególnych badań. Parametry środowiska przyjęte na potrzeby eksperymentów serii VI, zawiera tabela 5.16. Wartości parametrów algorytmu genetycznego przedstawia tabela 5.11.

Tabela 5.16. Wartości parametrów środowiska dla eksperymentów 1 - 2, serii VI

Nazwa parametru	Wartość
<i>Parametry symulacji</i>	
Rozmiar mapy	512 wm x 512 wm
Liczba pokoleń	1000
Liczba kroków symulacji	2000
Początkowa liczba kroków bez drapieżnika	250
<i>Parametry agentów ofiary</i>	
Liczba kopii agenta na początku symulacji	50
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	8
<i>Parametry agentów drapieżnika</i>	
Liczba kopii agenta w symulacji	2
Rodzaj sterownika	<i>RNN lub LSTM</i>
Liczba neuronów ukrytych sterownika	8

### Eksperyment 1 - Porównanie sterownika wykorzystującego sieć RNN oraz LSTM podczas ewolucji ofiar

W eksperymencie wykorzystano miarę średniej przeżywalności ofiar, gęstości i rozpiętości ich stada oraz liczby prób ataków drapieżnika. Sterownikiem obydwu gatunków (ofiari i drapieżników) jest naprzemienne sieć RNN oraz LSTM. Drapieżnicy zostali wyposażeni we wcześniej wyselekcjonowany genotyp (odpowiedni dla rodzaju sterownika) oraz wyłączeni z procesu dalszej ewolucji. Mechanizm komunikacji agentów ofiar i drapieżników nie został uwzględniony.

### Wyniki

Uzyskane wyniki zostały przedstawione na rysunkach 5.100 - 5.103, w formie wykresów średnich wartości wykorzystanych miar dla kolejnych pokoleń. Kolorem niebieskim i pomarańczowym przedstawiono rezultaty dla agenta ofiary wykorzystującego kolejno sterownik sieci RNN i LSTM oraz drapieżnika ze sterownikiem RNN, natomiast kolorem zielonym i czerwonym dla ofiary sterowanej siecią RNN i LSTM przy drapieżniku ze sterownikiem LSTM.

### Komentarz

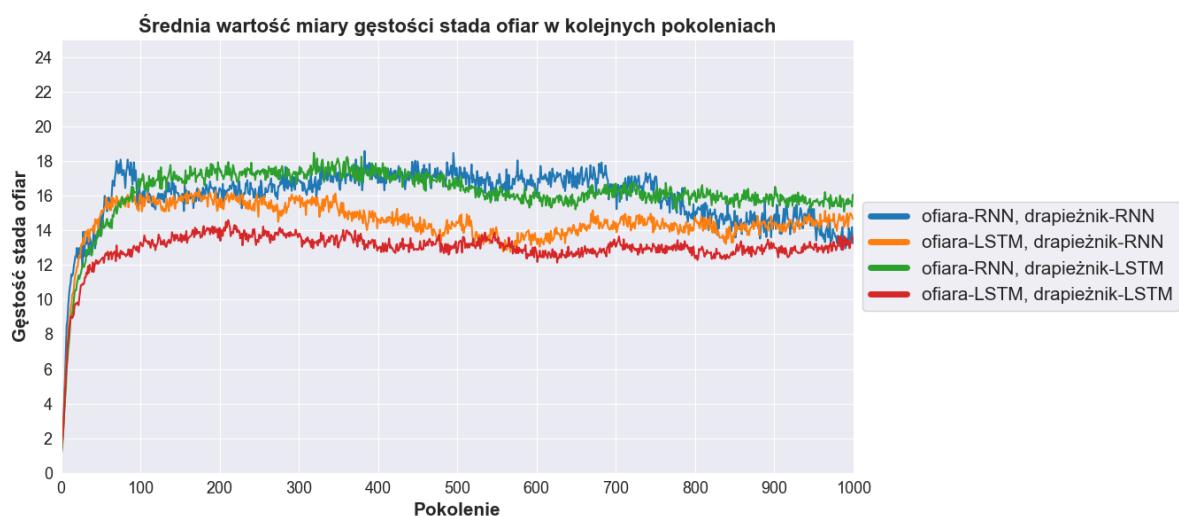
Pod względem średniej wartości przeżywalności ofiar, niezależnie od zastosowanego rodzaju sterownika, uzyskane wyniki są praktycznie takie same.

W przypadku gęstości stada można zauważyc, że ofiary sterowane siecią RNN nauczyły się formować gęstsze grupy, jednak różnica ta ulega zmianie w kolejnych pokoleniach, a pod koniec uzyskane wyniki są zbliżone. Te zmiany nie mają wpływu na osiąganą przeżywalność. Na wykresie średniej liczby prób ataków sytuacja wygląda podobnie, znów z niewielką przewagą po stronie ofiar wykorzystujących sterownik RNN.

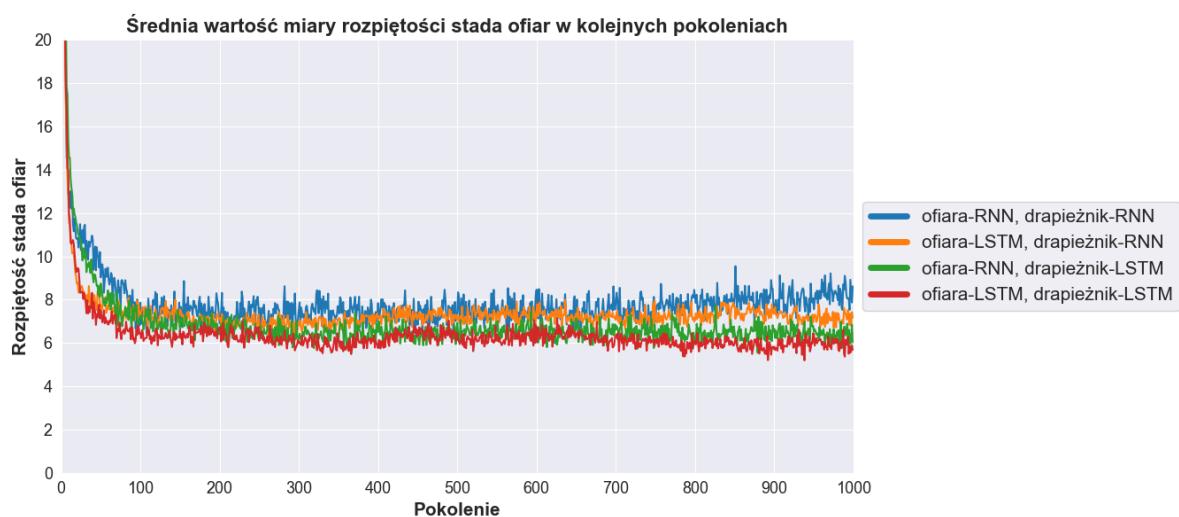
Podsumowując, ofiary sterowane siecią RNN osiągały nieznacznie mniejszą liczbę prób ataków, a formowane przez nie stada były nieco gęstsze. Różnice między sterownikami są jednak dość niewielkie, dlatego nie da się ostatecznie stwierdzić, który sterownik lepiej nadaje się w roli sztucznego mózgu agentów. Zwłaszcza biorąc pod uwagę fakt, że wykorzystywana w procesie ewolucji funkcja oceny agentów jest oparta na przeżywalności ofiar, a jej wartości (Rys. 5.100) są bardzo zbliżone niezależnie od rodzaju sterownika.



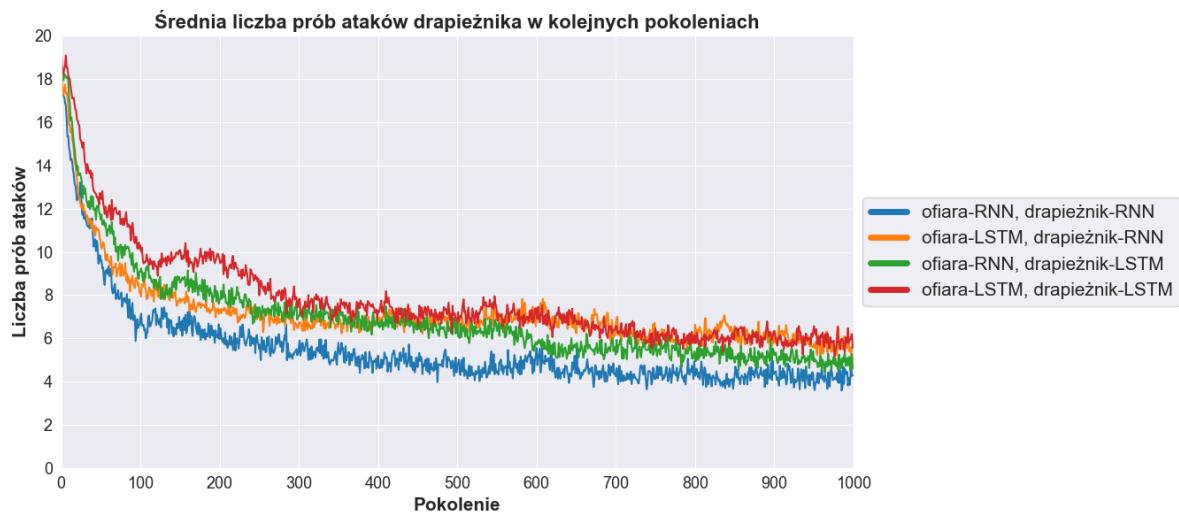
Rys. 5.100. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.



Rys. 5.101. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.  
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.



Rys. 5.102. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.  
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.



Rys. 5.103. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.

## Eksperyment 2 - Porównanie sterownika wykorzystującego sieć RNN oraz LSTM podczas ewolucji drapieżników

Do badań wykorzystano średnią miarę przeżywalności ofiar, średnią rozpiętość drapieżników oraz liczbę prób i udanych ataków przez nie przeprowadzonych. W roli sterownika obydwu agentów zastosowano naprzemiennie sieć RNN oraz LSTM. Agenty ofiar zostały wyposażone we wcześniej wyselekcjonowany genotyp (odpowiedni dla rodzaju sterownika), który posiada cechy odpowiedzialne za formowanie się stad oraz wyłączone z procesu dalszej ewolucji. Mechanizm komunikacji obydwu agentów został wyłączony.

### Wyniki

Uzyskane w ramach eksperymentu wyniki przedstawiono na rysunkach 5.104 - 5.107, w formie wykresów średnich wartości wykorzystanych miar w kolejnych pokoleniach. W kolorze niebieskim i pomarańczowym przedstawiono wyniki otrzymane dla agenta drapieżnika wykorzystującego kolejno sterownik sieci RNN i LSTM oraz ofiary ze sterownikiem RNN. Kolorem zielonym i czerwonym dla drapieżnika sterowanego siecią RNN i LSTM przy ofierze ze sterownikiem LSTM.

### Komentarz

Uzyskane wyniki pokazują, że w przypadku, kiedy ewolucji zostały poddane drapieżniki, różnice między dwoma sterownikami są znaczne. Średnia przeżywalność dla drapieżnika wyposażonego w sterownik RNN jest taka sama, niezależnie od sterownika ofiar, jednak dla drapieżnika ze sterownikiem LSTM, można zauważać znacznie wyższą przeżywalność ofiar, co oznacza, że ich ataki były mniej skuteczne.

Liczba prób ataków okazała się wyższa dla drapieżników sterowanych siecią RNN. Co ciekawe, liczba skutecznych ataków była większa oraz rosła znacznie szybciej dla ofiar sterowanych siecią LSTM, co może oznaczać, że stanowiły one łatwiejszy cel. W poprzednim eksperymencie to właśnie stado ofiar sterowanych siecią LSTM było mniej gęste.

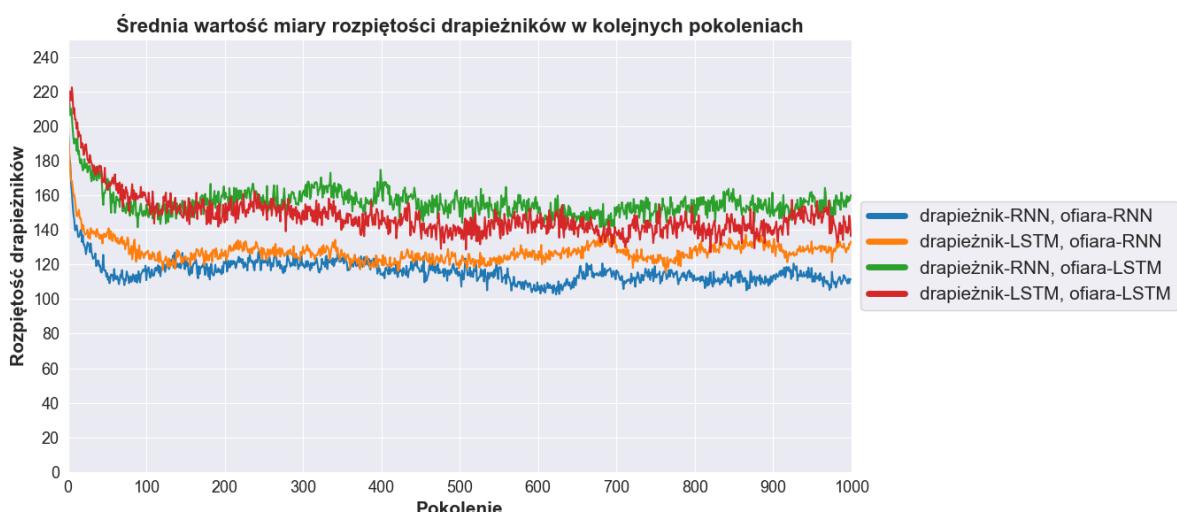
Słowem podsumowania, poprzedni eksperyment pokazał, że obydwa sterowniki były na zbliżonym poziomie, jednak z niewielką przewagą po stronie sieci RNN. W typ przypadku jednak, wyraźnie widać, że sterownik RNN prowadził do wykształcenia lepszych drapieżników, którzy skuteczniej polowali. Może to wynikać z różnicą w rozmiarze genotypu (dla sterownika LSTM

jest znacznie dłuższy), a tym samym większej liczby genów, których wartości należy prawidłowo dobrać. Ta różnica nie była aż tak widoczna w poprzednim eksperymencie, ponieważ polowanie na stada ofiar jest trudniejszym zadaniem niż formowanie stada. W tym drugim przypadku wystarczy, że ofiary nauczą się poruszać w stronę innych osobników własnego gatunku, natomiast polowanie wymaga zbliżenia się na niewielką odległość do celu, który jest bardziej zwinny.

Algorytm genetyczny wykorzystany w niniejszej pracy, a także w [12, 15, 16], jest dość prosty. Wykorzystanie bardziej zaawansowanego podejścia, na przykład algorytmu ewolucji różnicowej [4] (ang. *Differential Evolution*), który jest lepiej przystosowany do pracy na genotypie o wartościach rzeczywistych (a taki jest tu wykorzystany), umożliwiłoby skuteczniejsze uczenie i pozwoliło na wykorzystanie większego potencjału sieci LSTM.



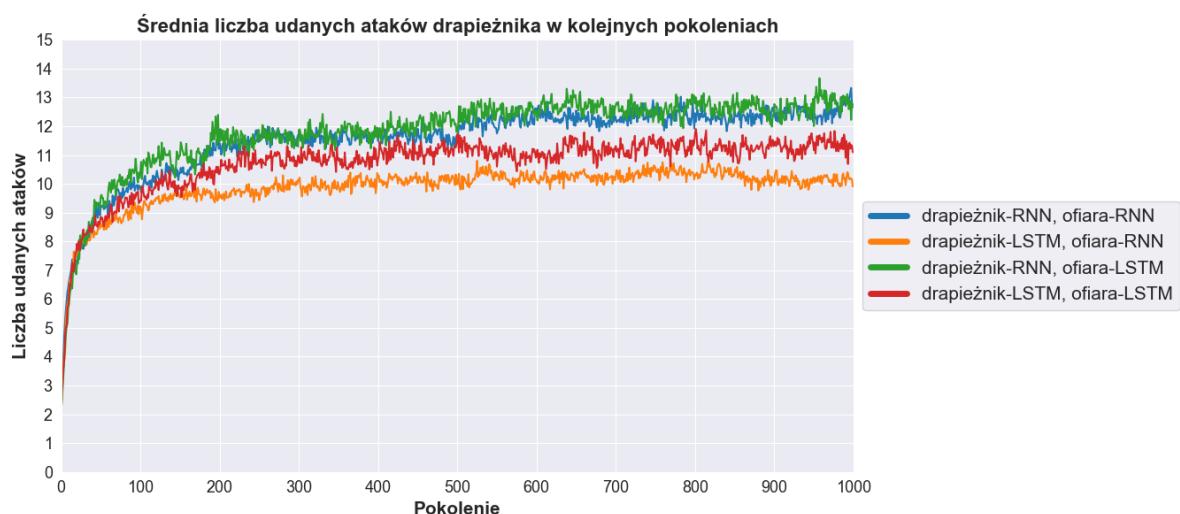
Rys. 5.104. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.  
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.



Rys. 5.105. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.  
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.



Rys. 5.106. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.



Rys. 5.107. Wykres średniej liczby udanych ataków drapieżnika w kolejnych pokoleniach.  
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.

## 6. PODSUMOWANIE BADAŃ EKSPERYMENTALNYCH

Wstępnie przeprowadzone badania, dotyczące doboru metod oraz parametrów algorytmu genetycznego wykazały, że najskuteczniejszą selekcją jest metoda rankingowa, polegająca na wyborze połowy najlepszych osobników z populacji. Metoda koła ruletki oraz turniejowa, które zapewniają większą różnorodność genetyczną populacji, dzięki dopuszczeniu do kolejnego pokolenia również słabszych osobników, wypadły w porównaniu znacznie słabiej.

Ze wszystkich przetestowanych metod mutacji, największy wpływ na skuteczność procesu ewolucji miała mutacja pojedynczego genu z prawdopodobieństwem 1% (wyższe wartości znacznie ją obniżały). Zastosowanie dodatkowych metod mutacji nie przyczyniło się do dużej poprawy uzyskiwanych wyników, a przy nieprawidłowym doborze wartości prawdopodobieństwa znacznie je pogarszało.

Z przebadanych metod krzyżowania, dla sterownika bazującego na sieci RNN najlepiej sprawdziło się podejście *jedno-punktowe*, natomiast w przypadku LSTM *dwu-punktowe na każdy segment*. Prawdopodobnie wynika to z różnic w ich wewnętrznej budowie. Sieć LSTM posiada cztery bramki, z których każda ma swój własny zestaw neuronów i wag, dlatego krzyżowanie przebiegające w obrębie każdej z nich przyniosło lepsze rezultaty. Istotne znaczenie miało również prawdopodobieństwo zajścia krzyżowania, które przy zbyt dużej wartości znacząco zmniejszało skuteczność uczenia.

Sieć LSTM wykazywała się większą odpornością na sposób inicjalizacji genotypu. Dla sterownika RNN, proces ewolucji przebiegał skuteczniej przy mniejszym stopniu początkowego wypełnienia genotypu. Wprowadzenie dodatkowej metody selekcji elitarnej nie przyniosło znaczącej poprawy uzyskiwanych wyników. Ostatecznie udało się uzyskać parametry, dzięki którym proces ewolucji zachodzi skuteczniej niż dla wartości zastosowanych w pracach [12, 15, 16].

Badania przeprowadzone nad liczbą ukrytych neuronów pokazały, że niezależnie od rodzaju zastosowanego sterownika, ofiary były w stanie poprawie wykorzystać efekt dezorientacji drapieżnika na swoją korzyść, co prowadziło do ewolucji zachowań stadnych, jednak większe wartości liczby neuronów powodowały spadek skuteczności uczenia się, a w konsekwencji do niższej przeżywalności ofiar. Przyczyną tego może być większy rozmiar genotypu, który rośnie wraz z liczbą neuronów sterownika. Badania dotyczące poszukiwania parametrów algorytmu genetycznego zostały przeprowadzone dla sterownika z 8 neuronami, co również mogło wpływać na wyniki.

Wprowadzenie do środowiska większej liczby drapieżników polujących na ofiary skutkowało proporcjonalnym spadkiem ich przeżywalności. Nie zaobserwowano zmian zachowania agentów, ani różnic w formowanych przez nie stadach, która byłaby zależna od liczby drapieżników. Już jeden drapieżnik stanowił presję wystarczającą do ewolucji zachowań stadnych.

Badania poświęcone ewolucji drapieżników pokazały, że byli oni w stanie nauczyć się jak skutecznie polować nawet na stada ofiar. Drapieżnicy nie musieli przemieszczać się po środowisku w poszukiwaniu celu, jak w przypadku ofiar rozproszonych, lecz wystarczyło znalezienie całej grupy ofiar. Spowodowało to kilkukrotny wzrost liczby prób ataków, a tym samym mimo efektu dezorientacji, który zmniejsza szansę na udany atak, drapieżnicy nadal byli w stanie dość skutecznie polować. Dzięki wizualizacjom utworzonym na podstawie przeprowadzonych simulacji,

zaobserwowano również formę współpracy drapieżników, które zazwyczaj atakowały tę samą grupę ofiar, nawet jeśli w środowisku było obecnych kilka stad. W dodatku, w pierwszej kolejności preferowały połowanie na pojedyncze osobniki lub grupy o mniejszej liczbie osobników.

W badaniach poświęconych konieczności zdobywania pożywienia przez ofiary, duży wpływ na wyniki miał sposób jego rozmieszczenia w środowisku. Dla pożywienia w formie rozproszonej, nie zauważono żadnych oznak zachowań stadnych. Rośliny dostępne były w całym środowisku, dlatego ofiary szybko i skutecznie nauczyły się jak je zdobywać. Pożywienie występujące w postaci skupisk stanowiło dla ofiar większe wyzwanie, co skutkowało zmniejszeniem średniej przeżywalności. W tym przypadku, średnia odległość między ofiarami była mniejsza, jednak wciąż nie nauczyły się jak skutecznie znaleźć pożywienie. Obecność pożywienia w formie pojedynczego skupiska, dodatkowo ukrytego w początkowej fazie symulacji, okazała się dla ofiar zbyt dużym wyzwaniem. Przetrwać były w stanie tylko nieliczne osobniki, którym przypadkiem udało się znaleźć źródło pożywienia.

Dodatkowa obecność drapieżników w środowisku, w którym ofiary musiały zdobywać pożywienie, jeszcze bardziej zmniejszyła ich przeżywalność. W niektórych przypadkach rozpiętość stada ofiar zmalała, co świadczy o tym, że podjęły próbę przeciwstawienia się atakom, jednak pogodzenie obydwu aspektów nie udało się.

Wprowadzenie mechanizmu komunikacji ofiar, zazwyczaj powodowało wzrost gęstości ich stada oraz średniej liczby prób ataków drapieżnika, co nie przekładało się jednak na zmianę przeżywalności. W niektórych przypadkach skutkowała nawet pogorszeniem uzyskiwanych wyników. Prawdopodobnie dodatkowe informacje, które wprowadza do wektora obserwacji nie zawsze były przydatne dla agentów, co często utrudniło proces ewolucji. Komunikacja drapieżników również nie przyniosła dla nich żadnej korzyści. Pozytywny wpływ mechanizmu komunikacji na przeżywalność ofiar, udało się zaobserwować w środowisku, w którym występowało 6 drapieżników. Ofiary znalazły również sposób na wykorzystanie umiejętności komunikacji przy pożywieniu występującym w postaci skupisk. Nauczyły się, że dźwięk wydawany przez pobliskie osobniki oznacza obecność roślin, a poruszając się w kierunku jego źródła będą w stanie zdobyć pożywienie. To zachowanie wpłynęło pozytywnie na przeżywalność ofiar.

Badania porównawcze sterownika wykorzystującego sieć RNN oraz LSTM pokazały, że jako "sztuczny mózg" agentów ofiar, obydwa sprawdzały się dość podobnie. Niewielka przewaga leżała po stronie ofiar sterowanych siecią RNN, ponieważ formowane przez nie stada były gęstsze oraz narażały się na mniejszą liczbę prób ataków drapieżnika. Sieć RNN poradziła sobie również zauważalnie lepiej w roli sterownika drapieżników, prowadząc do wykształcenia osobników, którzy polowali skuteczniej.

Przewaga sterownika RNN może wynikać z różnicy w rozmiarze genotypu, który dla sieci LSTM jest znacznie dłuższy, a tym samym posiada większą liczbę genów, którym wartości należą prawidłowo dobrać. Ponadto, algorytm genetyczny wykorzystany do przeprowadzenia procesu ewolucji jest dość prosty, dlatego może nie być wystarczający do wyuczenia znacznie bardziej złożonej sieci jaką jest LSTM i nie umożliwia wykorzystania jej pełnego potencjału.

## 7. ZAKOŃCZENIE

Głównym celem pracy było zbadanie użyteczności rekurencyjnej sieci neuronowej LSTM jako sterownika agentów ofiar i drapieżników, a także porównanie jej ze sterownikiem bazującym na klasycznej sieci rekurencyjnej RNN. Zbadano także wpływ komunikacji dźwiękowej oraz rodzaju zastosowanego sterownika na tworzenie się zachowań stadnych w środowisku przy obecnej presji drapieżnika oraz konieczności zdobywania pożywienia.

Wstępnie przeprowadzone badania posłużyły do dostrojenia parametrów algorytmu genetycznego, dzięki czemu proces ewolucji agentów przebiegał skuteczniej.

Udało się wykazać, że sterownik oparty na sieci LSTM sprawdza się w roli "sztucznego mózgu" agentów, jednak w porównaniu do sterownika RNN, wypadł nieco słabiej. Może to mieć związek z dość prostym algorymem genetycznym wykorzystanym do ewolucji agentów, który nie jest wystarczająco skuteczny i przez to nie umożliwia pełnego wykorzystania możliwości oferowanych przez sieć LSTM.

Obserwacje oraz wyniki badań pokazały, że ofiary wykształciły zachowania stadne w odpowiedzi na presję drapieżników. Nauczyły się formować stada, które pozwoliły przeciwstawić się ich atakom. Udało się również zaobserwować współpracę ofiar przy poszukiwaniu pożywienia występującego w formie skupisk. Grupowanie się zwiększało obszar, który mogły wspólnie przeszukać. Ofiary nauczyły się wykorzystać mechanizm komunikacji, dzięki któremu mogły informować pobliskie osobniki o znalezieniu źródła pożywienia, co pozytywnie wpłynęło na ich przeżywalność. Niestety, w środowisku przy jednoczesnej obecności drapieżników oraz konieczności zdobywania pożywienia, ofiarom nie udało się wykształcić zachowań, które umożliwiłyby skuteczne przetrwanie.

Ewolucja drapieżników udowodniła, że grupowanie się ofiar nie stanowi ostatecznej formy obrony przed ich atakami. Drapieżnicy byli w stanie nauczyć się jak zwiększyć szansę na udany atak, pomimo efektu dezorientacji, co przyczyniło się do skuteczniejszych polowań. Udało się również zaobserwować pewną formę współpracy, w której drapieżniki zazwyczaj atakowały tę samą grupę, nawet jeśli w środowisku ofiary utworzyły kilka mniejszych stad. W dodatku opracowały strategię, według której w pierwszej kolejności za cel obierają pojedyncze, oddalające się ofiary lub grupy o mniejszej liczbie osobników. Niestety, wprowadzenie mechanizmu komunikacji drapieżników nie przyczyniło się do poprawy skuteczności polowań czy wykształcenia bardziej złożonych zachowań.

Istnieje wiele potencjalnych kierunków na dalszy rozwój pracy. Jednym z nich jest rozbudowa obecnego mechanizmu komunikacji poprzez umożliwienie agentom różnych gatunków słyszeć wzajemnie wydawane dźwięki. Modyfikacja zmysłu wzroku mogłaby umożliwić dostrzeżenie kilku obiektów w każdym z segmentów pola widzenia.

Dalszy kierunek badań mógłby obejmować zmianę dotychczasowej oceny przystosowania agentów, na funkcję, która nagradzałaby za jak najdłuższe przetrwanie w środowisku. Takie podejście nie narzucałoby agentom konkretnego celu życia w środowisku oraz pozwoliło na większą swobodę ewolucji. Osobniki obydwu gatunków posiadałyby energię życiową, malejącą wraz z czasem, co wymusiłoby wykształcenie zachowań pozwalających na przetrwanie, np. zdobywanie pożywienia w postaci roślin czy polowanie na ofiary. Wówczas można przeprowadzić

badania przy równoczesnej ewolucji obydwu gatunków oraz obserwować w jaki sposób zmienia się ich zachowanie w czasie.

Idąc śladem pracy [22], można stworzyć bardziej realistyczne środowisko, uwzględniające takie aspekty jak naturalny proces rozmnążania przez poszukiwanie partnera, ochrona potomstwa czy energia życiowa organizmów i konieczność zdobywania pożywienia. Ewolucja obydwu gatunków agentów mogłaby przebiegać równolegle, przy wykorzystaniu bardziej złożonych sterowników, bazujących na rekurencyjnych sieciach neuronowych RNN i LSTM, w przeciwieństwie do zastosowanej w pracy [22], klasycznej sieci neuronowej, dzięki czemu agenty miałyby większe możliwości rozwoju.

Obiecująca wydaje się również perspektywa zastosowania bardziej zaawansowanych algorytmów ewolucyjnych, np. algorytm ewolucji różnicowej [4] (ang. *Differential Evolution*), który mogłyby znacznie uskutecznić i przyspieszyć proces nauki agentów, a tym samym przyczynić się do uzyskania ciekawszych i bardziej złożonych zachowań.

## BIBLIOGRAFIA

- [1] Adami, C., Moore, J., Dyer, F., Hintze, A., Olson, R., *Exploring the coevolution of predator and prey morphology and behavior*, Artificial Life Conference Proceedings. 2016.
- [2] Aguilar, W., Santamaría-Bonfil, G., Froese, T., Gershenson, C., *The past, present, and future of artificial life*, Frontiers in Robotics and AI. 2014.
- [3] Beer, R.D., Gallagher, J.C., *Evolving dynamical neural networks for adaptive behavior*, Adaptive Behavior. 1992.
- [4] Das, S., Suganthan, P., *Differential evolution: A survey of the state-of-the-art.*, IEEE Trans. Evolutionary Computation. 2011.
- [5] Demšar, J., Hemelrijk, C.K., Hildenbrandt, H., Bajec, I.L., *Simulating predator attacks on schools: Evolving composite tactics*, Ecological Modelling. 2015.
- [6] Goodfellow, I., Bengio, Y., Courville, A., *Deep Learning* (MIT Press, 2016).
- [7] Grand, S., *Artificial life*, Encyclopædia Britannica, inc. 2016.
- [8] Haupt, S.E., *Introduction to Genetic Algorithms* (Springer Netherlands, 2009).
- [9] ID., C., *Collective cognition in animal groups.*, Trends Cogn Sci. 2009.
- [10] Ioannou, C., Tosh, C., Neville, L., Krause, J., *The confusion effect - from neural networks to reduced predation risk*, Behavioral Ecology. 2008.
- [11] Kim, K.J., Cho, S.B., *A comprehensive overview of the applications of artificial life*, Artificial Life. 2006.
- [12] Markowska-Kaczmar, U., Marcinkowski, T., *Markov network versus recurrent neural network in forming herd behavior based on sight and simple sound communication*, Applied Soft Computing. 2020, tom 90, str. 106177.
- [13] Nielsen, M.A., *Neural networks and deep learning*. 2018.
- [14] Olah, C., *Understanding lstm networks*. 2015.
- [15] Olson, R.S., Hintze, A., Dyer, F.C., Knoester, D.B., Adami, C., *Predator confusion is sufficient to evolve swarming behaviour*, Journal of The Royal Society Interface. 2013.
- [16] Olson, R.S., Knoester, D.B., Adami, C., *Evolution of swarming behavior is shaped by how predators attack*, Artificial Life. 2016.
- [17] Reynolds, C.W., *Flocks, herds, and schools: A distributed behavioral model*, Computer Graphics, 21(4). 1987. Dostępny na <http://www.cs.toronto.edu/~dt/siggraph97-course/cwr87>.
- [18] Sherstinsky, A., *Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network*, Physica D: Nonlinear Phenomena. 2020, str. 132306.
- [19] Sloss, A.N., Gustafson, S., *Evolutionary algorithms review*, CoRR. 2019.
- [20] Sumpter, D., *The principles of collective animal behavior*, Philosophical transactions of the Royal Society of London. Series B, Biological sciences. 2006.
- [21] Ward, C.R., Gobet, F., Kendall, G., *Evolving collective behavior in an artificial ecology*, Artificial Life. 2001.
- [22] Werner, G.M., Dyer, M.G., *Evolution of herding behavior in artificial animals*. 1993.
- [23] Will, T.E., *Flock leadership: Understanding and influencing emergent collective behavior*, The Leadership Quarterly. 2016.
- [24] Wood, A., Ackland, G., *Evolving the selfish herd: Emergence of distinct aggregating strategies in an individual-based model*, Proceedings. Biological sciences / The Royal Society. 2007.
- [25] Xiaoying Wang, X.Z., *Pattern formation of a predator-prey model with the cost of anti-predator behaviors*, Mathematical Biosciences and Engineering. 2018.

## SPIS RYSUNKÓW

3.1.	Schemat sztucznej sieci neuronowej . . . . .	18
3.2.	Pętla sieci RNN oraz jej rozwinięcie w czasie . . . . .	20
3.3.	Schemat wewnętrznej budowy sieci LSTM . . . . .	21
4.1.	Schemat poglądowy przedstawiający pole wzroku agenta. . . . .	28
4.2.	Schemat budowy sterownika . . . . .	30
4.3.	Schemat genotypu sterownika RNN . . . . .	31
4.4.	Schemat genotypu sterownika LSTM . . . . .	31
5.1.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod selekcji dla sterownika RNN. . . . .	39
5.2.	Wykres średniej wartości przeżywalności ofiar w kolejnych generacjach. Porównanie prawdopodobieństwa pojedynczej mutacji genu dla sterownika RNN. . . . .	41
5.3.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod mutacji dla sterownika RNN. . . . .	42
5.4.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod mutacji dla sterownika RNN. . . . .	42
5.5.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod mutacji dla sterownika RNN. . . . .	42
5.6.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie najlepszych z przebadanych kombinacji mutacji dla sterownika RNN. . . . .	43
5.7.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie wartości prawdopodobieństwa krzyżowania dla sterownika RNN. . . . .	44
5.8.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod krzyżowania dla sterownika RNN. . . . .	45
5.9.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie najlepszych z przebadanych kombinacji dla sterownika RNN. . . . .	45
5.10.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie wartości początkowego wypełnienia genotypu $St_{init}$ dla sterownika RNN. . . . .	46
5.11.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie parametru $\sigma^2$ z rozkładu normalnego dla sterownika RNN. . . . .	47
5.12.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie rozmiaru selekcji elitarnej $N_{elit}$ dla sterownika RNN. . . . .	48
5.13.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod selekcji dla sterownika LSTM. . . . .	49
5.14.	Wykres średniej wartości przeżywalności ofiar w kolejnych generacjach. Porównanie prawdopodobieństwa pojedynczej mutacji genu dla sterownika LSTM. . . . .	50
5.15.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod mutacji dla sterownika LSTM. . . . .	51
5.16.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod mutacji dla sterownika LSTM. . . . .	52
5.17.	Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod mutacji dla sterownika LSTM. . . . .	52

5.18. Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie najlepszych z przebadanych kombinacji mutacji dla sterownika LSTM. . . . .	52
5.19. Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie metod krzyżowania dla sterownika LSTM. . . . .	54
5.20. Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie najlepszych z przebadanych kombinacji mutacji dla sterownika LSTM. . . . .	54
5.21. Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie wartości początkowego wypełnienia $St_{init}$ genotypu dla sterownika LSTM. . . . .	55
5.22. Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie parametru $\sigma^2$ z rozkładu normalnego dla sterownika LSTM. . . . .	56
5.23. Wykres średniej przeżywalności ofiar w kolejnych generacjach. Porównanie rozmiaru selekcji elitarnej $N_{elit}$ dla sterownika LSTM. . . . .	57
5.24. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika RNN. . . . .	60
5.25. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika RNN. . . . .	60
5.26. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika RNN. . . . .	60
5.27. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika RNN. . . . .	61
5.28. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika RNN. . . . .	62
5.29. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika RNN. . . . .	62
5.30. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika RNN. . . . .	62
5.31. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika RNN. . . . .	63
5.32. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika LSTM. . . . .	64
5.33. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika LSTM. . . . .	64
5.34. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika LSTM. . . . .	64
5.35. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika LSTM. . . . .	65
5.36. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika LSTM. . . . .	66
5.37. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika LSTM. . . . .	66
5.38. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika LSTM. . . . .	66
5.39. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach. Porównanie liczby ukrytych neuronów sterownika LSTM. . . . .	67
5.40. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach. Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników. . . . .	68
5.41. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach. Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników. . . . .	68
5.42. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach. Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników. . . . .	68

5.43. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników. . . . .	69
5.44. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN. . .	71
5.45. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN. . .	71
5.46. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN. . .	71
5.47. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN. . .	72
5.48. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN. . .	73
5.49. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN. . .	73
5.50. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN. . .	73
5.51. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika RNN. . .	74
5.52. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM. . .	75
5.53. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM. . .	75
5.54. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM. . .	75
5.55. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM. . .	76
5.56. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM. . .	77
5.57. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM. . .	77
5.58. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM. . .	77
5.59. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie wpływu liczby drapieżników obecnych w środowisku dla sterownika LSTM. . .	78
5.60. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników. . . . .	79
5.61. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników. . . . .	79
5.62. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.	
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników. . . . .	79
5.63. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie wpływu mechanizmu komunikacji ofiar dla obydwu sterowników. . . . .	80
5.64. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN. . .	82
5.65. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN. . .	82
5.66. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN. . .	82
5.67. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN. . .	84

5.68. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN. . . .	84
5.69. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika RNN. . . .	84
5.70. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM. . . .	85
5.71. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM. . . .	86
5.72. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM. . . .	86
5.73. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM. . . .	87
5.74. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM. . . .	87
5.75. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.	
Porównanie liczby drapieżników oraz mechanizmu komunikacji dla sterownika LSTM. . . .	88
5.76. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	90
5.77. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	91
5.78. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	91
5.79. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	91
5.80. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	93
5.81. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	93
5.82. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	93
5.83. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	94
5.84. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	95
5.85. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	95
5.86. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	95
5.87. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN. . . .	96
5.88. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM. . . .	97
5.89. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM. . . .	97
5.90. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM. . . .	97
5.91. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM. . . .	98
5.92. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM. . . .	99

5.93. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.	99
5.94. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.	99
5.95. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.	100
5.96. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.	101
5.97. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.	101
5.98. Wykres średniej rozpiętości stada ofiar w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.	101
5.99. Wykres średniej liczby zdobytego pożywienia przez ofiary w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika LSTM.	102
5.100. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.	104
5.101. Wykres średniej gęstości stada ofiar w kolejnych pokoleniach.	
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.	104
5.102. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.	
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.	104
5.103. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.	105
5.104. Wykres średniej przeżywalności ofiar w kolejnych pokoleniach.	
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.	106
5.105. Wykres średniej rozpiętości drapieżników w kolejnych pokoleniach.	
Porównanie sterownika wykorzystującego sieć RNN oraz LSTM.	106
5.106. Wykres średniej liczby prób ataków drapieżnika w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.	107
5.107. Wykres średniej liczby udanych ataków drapieżnika w kolejnych pokoleniach.	
Porównanie mechanizmu komunikacji ofiar oraz obecności drapieżnika dla sterownika RNN.	107

## **SPIS TABEL**

4.1.	Wartości wektora obserwacji wzrokowych dla sytuacji przedstawionej na Rys. 4.1 . . . . .	28
4.2.	Przykładowe wartości wektora obserwacji słuchowych agenta ofiary. . . . .	29
4.3.	Akcje agentów zależnie od wartości $A_i$ . . . . .	29
4.4.	Wartości parametrów środowiska. . . . .	36
5.1.	Parametry algorytmu genetycznego dla eksperymentu 1 serii I . . . . .	39
5.2.	Parametry algorytmu genetycznego dla eksperymentu 2 serii I . . . . .	40
5.3.	Parametry algorytmu genetycznego dla eksperymentu 3 serii I . . . . .	43
5.4.	Parametry algorytmu genetycznego dla eksperymentu 4 serii I . . . . .	45
5.5.	Parametry algorytmu genetycznego dla eksperymentu 5 serii I . . . . .	47
5.6.	Parametry algorytmu genetycznego dla eksperymentu 6 serii I . . . . .	49
5.7.	Parametry algorytmu genetycznego dla eksperymentu 7 serii I . . . . .	49
5.8.	Parametry algorytmu genetycznego dla eksperymentu 8 serii I . . . . .	53
5.9.	Parametry algorytmu genetycznego dla eksperymentu 9 serii I . . . . .	54
5.10.	Parametry algorytmu genetycznego dla eksperymentu 10 serii I . . . . .	56
5.11.	Ostatecznie wybrane parametry algorytmu genetycznego dla obydwu sterowników. . . . .	57
5.12.	Wartości parametrów środowiska dla eksperymentów 1 - 4, serii II . . . . .	58
5.13.	Wartości parametrów środowiska dla eksperymentów 1 - 4, serii III . . . . .	70
5.14.	Wartości parametrów środowiska dla eksperymentów 1 - 4, serii IV . . . . .	81
5.15.	Wartości parametrów środowiska dla eksperymentów 1 - 6, serii V . . . . .	89
5.16.	Wartości parametrów środowiska dla eksperymentów 1 - 2, serii VI . . . . .	103

## **DODATKI**

### **ZAWARTOŚĆ PŁYTY DVD**

Na płycie DVD dołączonej do pracy znajdują się następujące pliki:

- *Symulacja.zip* - skompresowany plik zawierający kod źródłowy programu oraz pliki niezbędne do uruchomienia symulacji wirtualnego środowiska.
- *Dane.zip* - skompresowany plik zawierający wyniki wszystkich eksperymentów przeprowadzonych w ramach pracy, w formie plików formatu *.csv*.
- *Genotyp.zip* - skompresowany plik zawierający wykorzystywane w trakcie badań, wcześniej wyselekcyjowane genotypy agentów, zapisane w plikach formatu *.pkl*.
- *Wizualizacje.zip* - skompresowany plik zawierający wizualizacje wybranych symulacji przeprowadzonych w ramach eksperymentów, zapisane w plikach formatu *.mp4*.
- *Instrukcja.pdf* - instrukcja zawierająca informacje dotyczące komplikacji, uruchomienia oraz dostosowania parametrów symulacji.

### **WIZUALIZACJE**

W ramach eksperymentów, na podstawie przeprowadzonych symulacji zostały wykonane wizualizacje, które można znaleźć na płycie DVD dołączonej do pracy lub na stronie internetowej pod adresem: [http://bit.ly/praca\\_magisterska\\_adrian\\_slimak](http://bit.ly/praca_magisterska_adrian_slimak).