

Wybrane strategie walki z przeciwnikiem w turowej grze strategicznej

Adam DĄBROWSKI*
Magdalena TOPCZEWSKA*

1. Wprowadzenie

Gra komputerowa to oprogramowanie, którego idea wywodzi się z gier planszowych i fabularnych oraz umożliwia użytkownikowi przede wszystkim rozrywkę [6]. Często wymaga jednak także rozwiązywania skomplikowanych zadań logicznych [5].

Jednym z popularnych rodzajów gier jest strategiczna gra turowa, w której scenariusz rozgrywki podzielony jest na kilka tur. W każdej turze gracz może wykonać określone ruchy, takie jak przemieszczanie, atakowanie czy budowanie. Na uzyskane wyniki w kolejnych turach mają wpływ ruchy wykonane w turach poprzedzających, więc gracz w celu otrzymania najlepszych wyników powinien opracować strategię grania, którą będzie dostosowywał i modyfikował w zależności od ruchów przeciwnika.

Tematem niniejszego opracowania jest zaprezentowanie oraz eksperymentalne porównanie wybranych podejść strategii gry turowej w przypadku walki przeciw sobie dwóch armii. Zaprezentowane zostaną takie statystyki, jak: procent wygranych gier, procent pozostałej po rozgrywce armii, liczba potrzebnych do zakończenia rozgrywki ruchów czy optymalność wykorzystania zasobów.

2. Opis gry

Podstawą przeprowadzenia opisanych w kolejnych rozdziałach eksperymentów jest strategiczna gra turowa stworzona przez autora – Adama Dąbrowskiego – oraz Sylwię Małyszę [3, 8]. Zaprojektowano oraz stworzono świat, plansze, modele i struktury. Podstawową funkcjonalnością jest możliwość przeprowadzania potyczek pomiędzy dwoma graczami, którzy pełnią w grze rolę dowódców dwóch armii. Każdy z dowódców posiada swoje wojsko, składające się z kilku rodzajów jednostek o różnej sile. Armia podzielona jest na oddziały, a każdy oddział składa się z ustalonej liczby jednostek tego samego typu. Jednostki zostają rozlokowane na dostępnych polach planszy. Jedna figurka jednostki reprezentuje cały oddział.

Wśród jednostek armii opisywanej gry wyróżnić można (rys. 1):

- kusznika – jednostkę dystansową, czyli wykonującą ataki na odległość, słabo mobilną, posiadającą niewielką odporność na ataki fizyczne i magiczne (rys. 1a);

* Politechnika Białostocka

- tarczownika – jednostkę walczącą w zwarciu, czyli atakującą wyłącznie jednostki na sąsiadujących polach lub polach, do których może się przemieścić, przeciętnie mobilną, odporną na ataki fizyczne, ale nie na ataki magiczne (rys. 1b);
- maga – jednostkę dystansową, zadającą duże obrażenia magiczne, przeciętnie mobilną, z niską wytrzymałością oraz słabą odpornością na ataki fizyczne (rys. 1c);
- kawalerię – jednostkę walczącą w zwarciu, ale posiadającą duży zasięg, mającą bardzo dużą odporność na obrażenia fizyczne i magiczne, stanowiącą najsilniejszy rodzaj jednostki w grze (rys. 1d);
- katapultę – jednostkę dystansową zadającą duże obrażenia fizyczne, bardzo mało mobilną, bardzo odporną na obrażenia fizyczne, natomiast słabo odporną na obrażenia magiczne (rys. 1e).

a) kusznik

b) tarczownik

c) mag

d) kawaleria

e) katapulta



RYS. 1. Obrazy jednostek na planszy oraz ich symbole

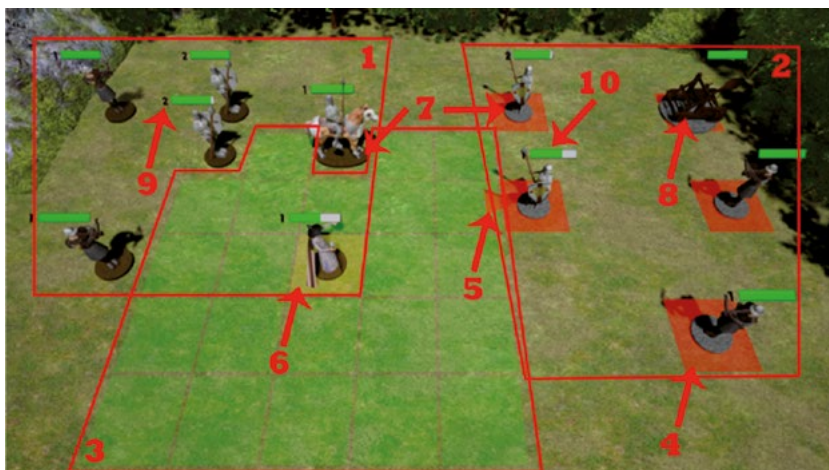
FIG. 1. Images of units on the board and their symbols

ŹRÓDŁO: opracowanie własne.

SOURCE: own elaboration.

Na rysunku 1 oprócz wyglądu figur na planszy zaprezentowane zostały także ich symbole używane zarówno w samej grze, jak i w dalszej części pracy.

W trakcie przeprowadzania bitwy pomiędzy dwoma armiami na ekranie komputera widoczne są dodatkowe informacje, ułatwiające graczowi podejmowanie decyzji (rys. 2). Są to m.in.: obydwie armie (rys. 2, zaznaczenie 1 i 2) oraz kolory ich platform (rys. 2, zaznaczenie 7); oddział, który aktualnie wykonuje ruch (rys. 2, zaznaczenie 6); podświetlenie pól, na które aktualny gracz może przemieścić swoje jednostki (rys. 2, zaznaczenie 3); pola, które mogą być zaatakowane strzałem (rys. 2, zaznaczenie 4); strzałki wskazujące obszar, który może zostać zaatakowany przez przeciwnika (rys. 2, zaznaczenie 5). Nad każdą figurą (np. rys. 2, zaznaczenie 8) przedstawiona jest dodatkowo liczba oznaczająca ilość jednostek w danym oddziale (rys. 2, zaznaczenie 9) oraz pasek życia obrazujący informację o ilości ran, jakie odniósł dany oddział (rys. 2, zaznaczenie 10).



RYS. 2. Zrzut ekranu wykonany podczas bitwy

FIG. 2. Screenshot during the battle

ŹRÓDŁO: opracowanie własne.

SOURCE: own elaboration.

Każdy gracz w trakcie swojej kolejki wykonuje jeden ruch. Pojedynczy ruch gracza polega na:

- przemieszczeniu jednostek na inne pole,
- przemieszczeniu jednostek wraz z atakiem na przeciwnika,
- ataku na przeciwnika bez przemieszczenia jednostek.

W przypadku przemieszczenia oddziałów i ataku w zwarciu można zawsze spodziewać się od przeciwnika kontrataku.

3. Algorytmy dotyczące strategii gry

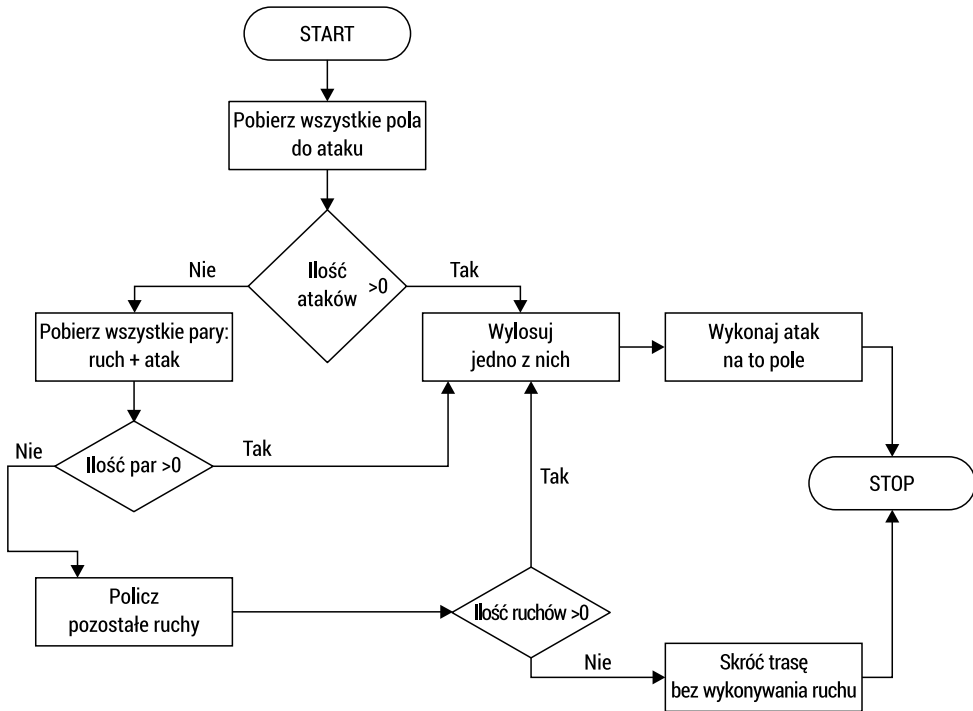
Wersja podstawowa stworzonej gry umożliwia odbycie pojedynków pomiędzy prawdziwymi zawodnikami. Istnieje także możliwość grania z komputerem, czyli z armią, która nie należy do żadnego z graczy. Aby to osiągnąć, gra została wyposażona w dodatkowy moduł odpowiedzialny za wykonywanie ruchów jednostkami.

Podstawowym problemem jest właściwy dobór ruchu spośród kilku, kilkunastu lub kilkudziesięciu możliwości. Istnieje cały szereg metod, które można wykorzystać przy implementacji sposobu poruszania się oddziałów w grze [5]. W tej pracy zaimplementowano oraz porównano trzy podejścia:

- losowe,
- najbliższego przeciwnika,
- oceny stanu na polu bitwy.

3.1. Algorytm losowy (AL)

Algorytm losowy należy do rodziny algorytmów prostych i prezentuje najprostsze podejście, będące punktem odniesienia do pozostałych algorytmów. W sposób losowy, z puli wszystkich dostępnych w danej rozgrywce ruchów, wybierany jest jeden. Schemat blokowy metody przedstawiony został na rysunku 3.



RYS. 3. Schemat blokowy algorytmu losowego

FIG. 3. Block diagram of a random algorithm

ŹRÓDŁO: opracowanie własne.

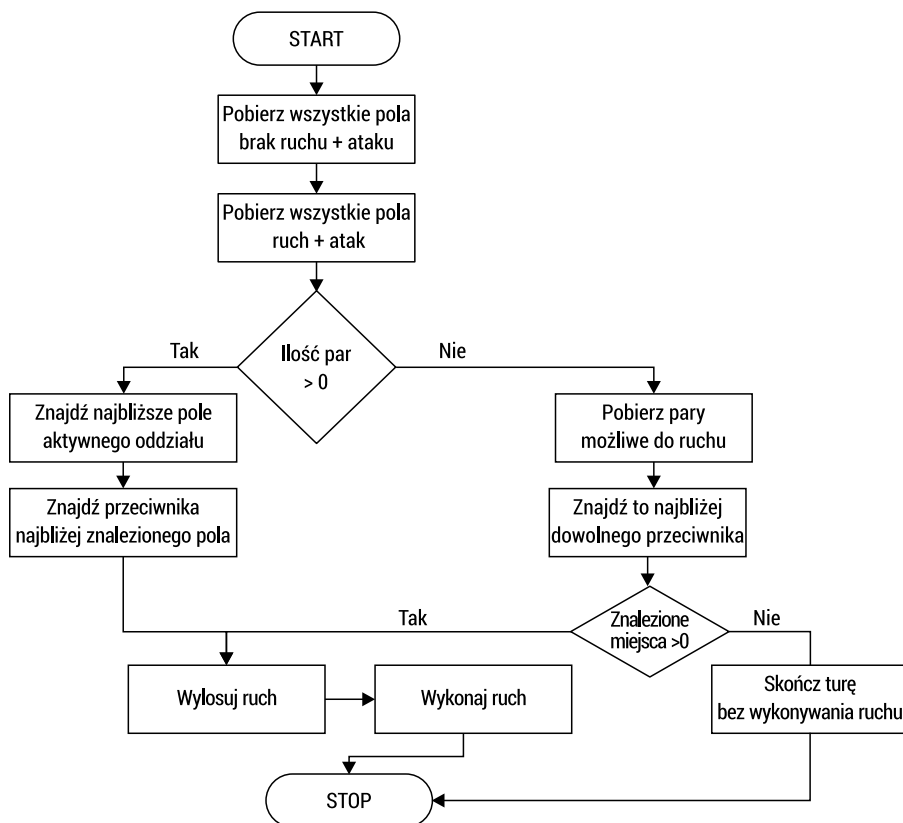
SOURCE: own elaboration.

Z przeprowadzonych eksperymentów wynika, że podejście to daje bardzo przypadkowe ruchy, w związku z czym jednostki mogą np. cofać się zamiast atakować. W konsekwencji przypadkowe działania mogą prowadzić do wydłużenia czasu trwania bitwy. W związku z tym wprowadzono modyfikację polegającą na wymuszeniu ataku, jeśli istnieje taka możliwość, eliminując opisany powyżej problem.

Dodatkowo, by uniknąć ataku odwetowego, preferowany jest atak dystansowy. W przypadku gdy w ogóle nie ma możliwości ataku, jednostki przemieszczane są w sposób losowy. Jeśli brak jest możliwości ruchu, co jest bardzo mało prawdopodobne, wówczas nie jest wykonywany żaden ruch.

3.2. Algorytm najbliższego przeciwnika (ANP)

Drugim analizowanym podejściem wyboru ruchu w grze jest atakowanie lub zbliżanie się do przeciwnika, który jest w najbliższej odległości od jednostek gracza. W każdej iteracji dla każdego oddziału obliczane są odległości euklidesowe do każdego oddziału przeciwnika, a następnie wybierany jest oddział, który znajduje się najbliżej. Dzieje się tak w przypadku, gdy konieczne jest przeprowadzenie ataku w zwarciu.



RYS. 4. Schemat blokowy algorytmu najbliższego przeciwnika

FIG. 4. Block diagram of the nearest enemy algorithm

ŹRÓDŁO: opracowanie własne.

SOURCE: own elaboration.

Istotną zaletą obydwu zaprezentowanych algorytmów jest ich prostota, która jednocześnie stanowi jednak ich ogromną wadę. Granie z przeciwnikiem wybierającym ruchy w sposób losowy lub wybierającym jedynie umiejscowione najbliżej siebie jednostki przeciwnika powoduje, że takiego gracza można pokonać w prosty sposób.

Ruchy w grze stają się przewidywalne, przez co cała rozgrywka przestaje być ciekawa dla graczy. W związku z tym zaproponowany został algorytm, który do wyboru kolejnego ruchu w grze będzie wykorzystywał ocenę aktualnej sytuacji na polu bitwy.

3.3. Algorytm oceny stanu na polu bitwy (AOSnPB)

Algorytm oceny stanu na polu bitwy wykorzystuje ideę gry w szachy, gdzie przed podjęciem decyzji o kolejnym ruchu, analizuje się dostępne możliwości, by wybrać optymalne rozwiązanie. W szachach wykorzystywany jest tzw. system oceny pozycji [1].

Do szacowania stanu na polu rozpatrywane są w grze dwa rodzaje wartości armii: materialna i niematerialna. Na wartość materialną składają się liczba oddziałów, jakie posiada gracz oraz siła jednostek, czyli ich typ. Wartość niematerialna związana jest natomiast z pozycjami, na jakich rozmieszczone są oddziały na planszy, co może mieć bezpośredni wpływ na wynik rozgrywki. Następnie, dla każdej możliwości ruchu obliczany jest i zapamiętywany stosunek wartości obu armii. Dysponując wszystkimi wynikami można ostatecznie wybrać rozwiązanie dające największą przewagę nad przeciwnikiem. Schemat blokowy algorytmu zaprezentowany został na rysunku 5.

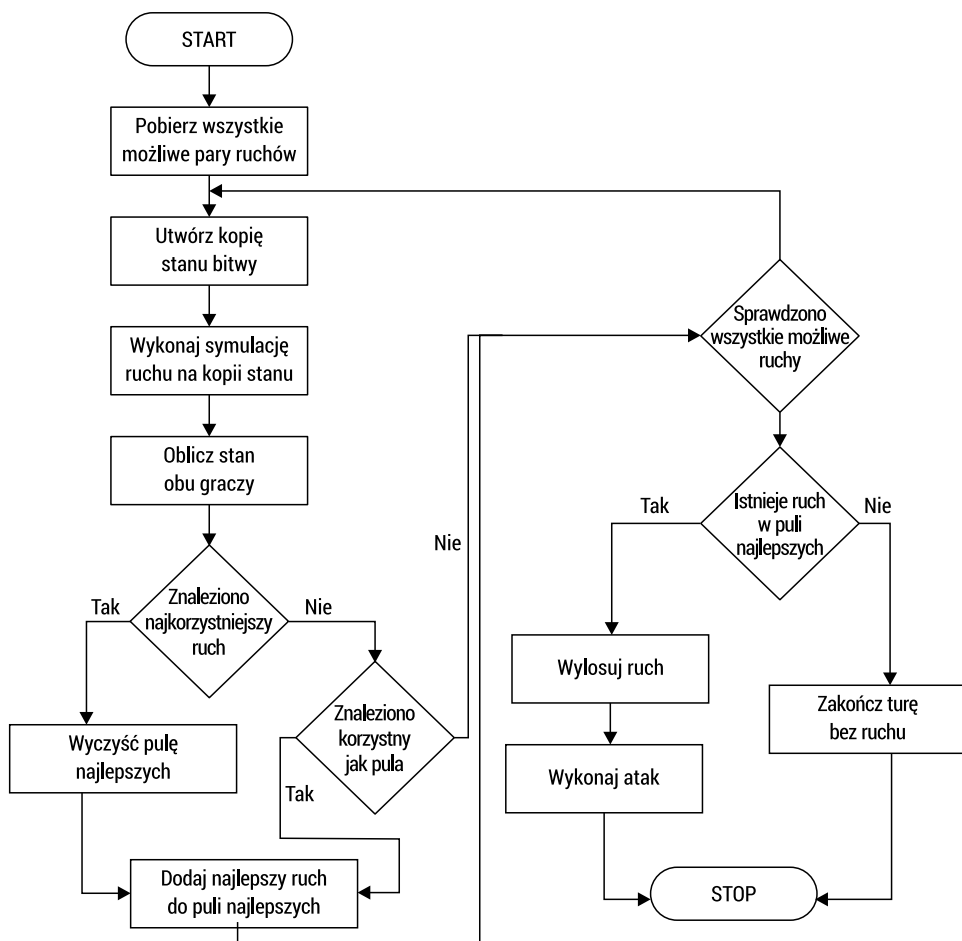
W celu stworzenia matematycznego oszacowania oceny stanu bitwy wykorzystane zostały zarówno składowa materialna, jak i niematerialna. Jednak ze względu na trudność napotkaną przy przeliczaniu wpływu liczby jednostek i ich siły na ocenę, wartość materialna została przedstawiona w postaci dwóch pochodnych składowych: siły ataku oraz wytrzymałości armii.

Ostatecznie, składowymi funkcji oceny zostało siedem czynników:

$$S = \{W_a, W_w, W_k, D_z, D_a, P, K_r\} \quad (1)$$

gdzie:

- W_a – wartość ataku – wartość wpływu na obrażenia zadawane oddziałom przeciwnika; wyższa wartość oznacza większe obrażenia przeciwnika;
- W_w – wartość wytrzymałości – punkty życia każdej jednostki zredukowane podczas doznawania obrażeń; im więcej punktów posiada jednostka, tym jest bardziej wytrzymała i może dłużej utrzymać się na polu walki;
- W_k – wartość kontrataku – atak obrońcy, z którym spotyka się grający po wykonaniu swego ruchu w zwarciu; niska wartość parametru powoduje brak wpływu na ocenę, a wysoka wartość może wpłynąć na wybór ruchu, w którym preferowany będzie atak i nastąpi niekorzystny dla gracza kontratak;
- D_z – odległość od przeciwnika jednostek walczących w zwarciu – jednostki walczące w zwarciu nabierają większego znaczenia w grze w przypadku, gdy ich przeciwnicy znajdują się w odległości umożliwiającej atak; mniejsza odległość między członkami własnej armii oznacza bliższą odległość od przeciwnika, a co za tym idzie – lepszy stan armii na polu bitwy;



RYS. 5. Schemat blokowy algorytmu oceny stanu na polu bitwy

FIG. 5. Block diagram of the battle evaluation algorithm

ŹRÓDŁO: opracowanie własne.

SOURCE: own elaboration.

- D_d – odległość od przeciwnika jednostek dystansowych – w przypadku jednostek dystansowych preferowana jest ich większa odległość od jednostek przeciwnika; w algorytmie nie są brane pod uwagę odległości od strzelców przeciwnika, gdyż jednostki te mogą atakować przeciwnika z każdej pozycji;
- P_p – przewaga przewagi – wpływ rozmieszczenia oddziałów i ich siły na planszy; preferowanym ustawieniem jest duża odległość od silnych oddziałów przeciwnika, a mała od słabych;
- K_r – kolejność ruchu – wpływ kolejności wyboru jednostki na możliwość ataku i szybkiego wyeliminowania z gry jednostek przeciwnika.

Poprawność implemetacji wszystkich składowych została sprawdzona eksperymentalnie poprzez przetestowanie przygotowanych scenariuszy bitew. Ustawiając wartości sześciu z siedmiu wag poszczególnych składowych na zero, sprawdzono również działanie każdej składowej osobno.

Wadą zaprezentowanego algorytmu jest konieczność wyliczenia wartości oceny dla każdej możliwości ruchu w danym momencie gry, dlatego, w porównaniu z poprzednimi, algorytm ten jest wolniejszy. Czas obliczeń jest jednak akceptowalny i niezauważalny dla gracza. Niewątpliwą zaletą jest możliwość porównania ruchów, które graczowi wydają się równie dobre oraz znajdowanie ruchów, które nie są oczywiste, a dają leszy rezultat.

Na zachowanie algorytmu można dodatkowo wpływać poprzez zmianę wag w_i funkcji oceny:

$$O = w_1 W_a + w_2 W_w + w_3 W_k + w_4 D_z + w_5 D_d + w_6 P_p + w_7 K_r \quad (2)$$

gdzie: w_i oznacza i -tą wagę.

Poszczególne składowe funkcji oceny (2) mogą mieć różny wpływ na ostateczny wynik ze względu na różne rzędy wielkości, dlatego pierwszą dodatkową opcją może być normalizacja składowych wyrazów. Można także wpływać na wartość wag, niwelując różnice we wpływie lub tak dobierając wartości, by wyróżnić pewne pożądane zachowania w trakcie przeprowadzania potyczki.

Do znalezienia zestawu wag można wykorzystać różne rozwiązania [4], w tym sieci neuronowe [7]. Problemem jest tu jednak jest tu brak danych – zbioru uczącego, czyli historii gier wraz z ustawieniami, na których można byłoby uczyć sieć. Potrzebne jest więc rozegranie szeregu gier, przy czym losowe ich generowanie jest zbyt kosztowne do przeprowadzenia eksperymentów. Ustalono, że w pierwszym kroku należy przetestować różne zestawy wag, zmieniając je z dużym krokiem, natomiast by wyniki były miarodajne, niezbędna jest odpowiednia liczba rozegranych gier. W celu doboru wag ograniczono się do trzech ustawień oddziałów na polu bitwy (rys. 6a, b i c) oraz przeszukiwania przestrzeni wartości w sposób opisany poniżej.

W pierwszej iteracji wartość wagi wybierana jest na podstawie informacji o tym, w jaki sposób odpowiadająca składowa ma wpływ na wynik gry, szacując, czy wartość wagi powinna być małą czy dużą liczbą. Na tej podstawie przypisuje się wodom odpowiednio wartość minimalną lub maksymalną. Drastyczne ograniczenie zbioru wartości wag powoduje, że do przetestowania otrzymano 128 kombinacji wag. Po sprawdzeniu, która z dwóch wartości wpływa na lepszy ogólny wynik, przedział wartości wagi jest następnie dzielony na dwie połowy, a do dalszych poszukiwań wybiera się połowę, w której znajduje się wartość dająca lepsze rezultaty. Na tej samej zasadzie powtarzano kolejne iteracje algorytmu poszukiwań, zawężając przedziały wartości wag. Do porównań znalezionych rozwiązań oraz wyboru najlepszego zestawu wag wykorzystany został przypadek, gdy wszystkie wagi były równe. Dla każdej kombinacji wag rozegrano 500 gier, z czego jedna rozgrywka trwała ok. 1 s. Łącznie przeprowadzono pół miliona symulacji, które trwały 120 godzin. Szczegóły doboru wag algorytmu oceny stanu na polu bitwy przedstawione zostały w [2].

4. Ekperymentalne porównanie algorytmów

Do przeprowadzenia porównań działania trzech algorytmów strategii walki z przeciwnikiem oraz oszacowania różnic w wynikach rozgrywek wykorzystano sześć ustawień oddziałów armii na planszy (rys. 6):

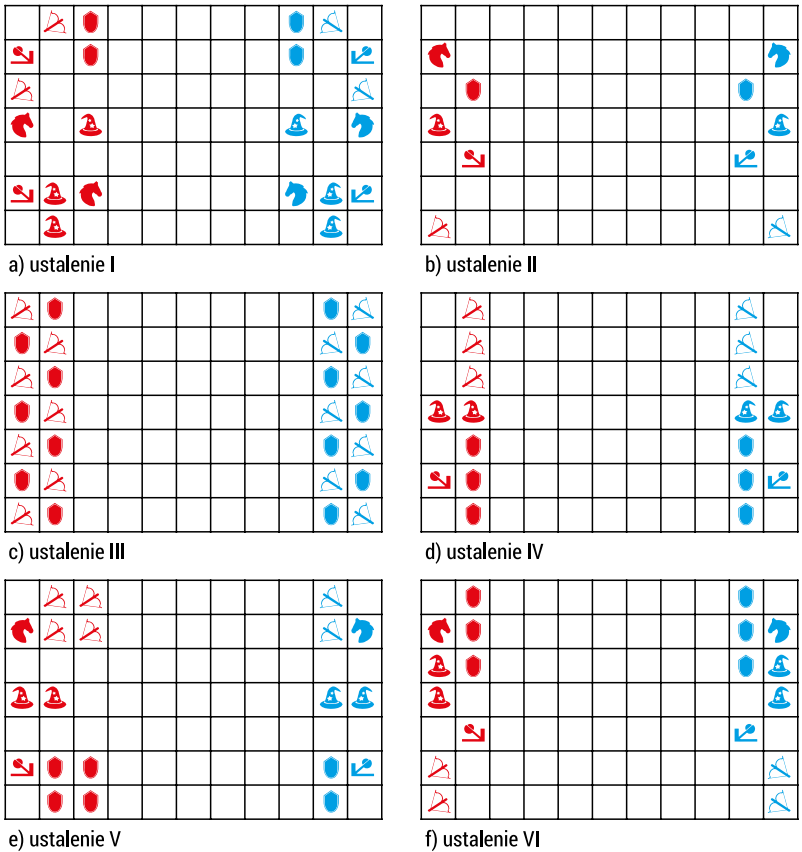
- ustawienie I – obydwaj gracze dysponują dwoma oddziałami każdego typu, rozmieszczonymi symetrycznie, choć w sposób nieuporządkowany; obecne oddziały magiczne;
- ustawienie II – obydwaj gracze dysponują pojedynczymi oddziałami każdego typu, rozmieszczonymi symetrycznie, choć w sposób nieuporządkowany; obecne oddziały magiczne;
- ustawienie III – obydwaj gracze dysponują oddziałami tylko dwóch typów: walczącymi w zwarciu i na dystans, rozmieszczonymi symetrycznie, zajmującymi dwa pełne rzędy planszy;
- ustawienie IV – obydwaj gracze dysponują oddziałami różnego typu z przewagą oddziałów o mniejszej sile (kusznicy, tarczownicy), rozmieszczonymi symetrycznie; obecne oddziały magiczne;
- ustawienie V – obydwaj gracze dysponują różnymi oddziałami różnego typu o nierównoważonych siłach; armia gracza z lewej strony planszy ma więcej oddziałów; rozmieszczenie niesymetryczne, obecne oddziały magiczne;
- ustawienie VI – obydwaj gracze dysponują różnymi oddziałami różnego typu o nierównoważonych siłach; armia gracza z lewej strony planszy ma więcej oddziałów – po cztery, gracza po prawej po trzy; rozmieszczenie symetryczne; obecne oddziały magiczne.

Podczas przeprowadzania potyczek gromadzone były podstawowe oraz rozszerzone statystyki gry, które zaprezentowane zostały w tabelach 1, 2 i 3 z wynikami przeprowadzonych bitew: procent wygranych gier (P_w), procent pozostałej armii (P_a), liczba ruchów (N_r) oraz optymalność ataków (O_a). Optymalność ataków O_a liczona była jako stosunek obrażeń zadanych przez oddziały do obrażeń, jakie te oddziały mogłyby zadać. Jeśli dziesięcioosobowy oddział kuszników może w jednym ataku pokonać czterech kuszników z armii przeciwnej, to w przypadku ataku na oddział dwóch kuszników wykorzystuje swój potencjał w 50%. Parametr ten w niewielu przypadkach osiąga wartość 100%. Wszystkie z zaprezentowanych statystyk są wartościami średnimi z uruchomionych 500 gier dla każdego typu eksperymentu.

Eksperyment 1

Eksperyment pierwszy opisuje przypadki ustawień, w których obydwie armie mają równoważne siły. Dotyczy to ustawień: I, II, III i IV. W tabeli 1 znajdują się uzyskane wynikowe statystyki przeprowadzonych eksperymentów i symulacji walk. Prezentowane są wyniki symulacji dla armii wykorzystującej do wyboru ruchu algorytm

najbliższego przeciwnika przeciw armii wykorzystującej do wyboru ruchu algorytm losowy, wyniki symulacji dla armii wykorzystującej algorytm oceniający stan na polu bitwy przeciw armii stosującej algorytm losowy oraz wyniki bezpośredniej walki, gdy używane są algorytmy ANP oraz AOSnPB.



RYS. 6. Wybrane ustawienia oddziałów

FIG. 6. Selected army units arrangements

ŹRÓDŁO: opracowanie własne.

SOURCE: own elaboration.

Analizując wyniki uzyskane w przypadku ustawienia I (tab. 1), można stwierdzić, że najlepsze rezultaty uzyskano dla algorytmu oceniającego stan na polu bitwy: procent wygranych gier był na poziomie 90,40% w walce wykorzystującej przez przeciwnika algorytm losowy (76,40%), a średni procent zachowanej armii wyniósł 32,57% w porównaniu z 21,12% dla algorytmu losowego. Również optymalność wykorzystania zasobów była o 1.79% wyższa w stosunku do algorytmu losowego. W bezpośrednim starciu większy procent wygranych uzyskał natomiast algorytm AOSnPB: 73,40% przy 20,90% zachowanej armii.

TAB. 1. Wyniki przeprowadzonych potyczek przy czterech wybranych ustawieniach

TAB. 1. Results of the battles for four selected army units arrangements

Ustawienie	Walka		P_w [%]	P_a [%]	N_r	O_a [%]
I	przeciw AL	ANP	76,40	21,12	18,46	67,90
		AOSnPB	90,40	32,57	18,22	69,69
	bezpośrednia	ANP	26,60	4,22	12,75	72,34
		AOSnPB	73,40	20,90	20,48	68,74
II	przeciw AL	ANP	0,00	0,00	113,47	69,82
		AOSnPB	100,00	47,03	139,22	68,05
	bezpośrednia	ANP	0,00	0,00	80,08	77,28
		AOSnPB	100,00	50,13	134,29	68,23
III	przeciw AL	ANP	76,40	16,10	49,91	68,64
		AOSnPB	96,60	29,25	46,93	71,83
	bezpośrednia	ANP	3,40	0,27	32,35	72,76
		AOSnPB	96,60	29,42	48,03	70,82
IV	przeciw AL	ANP	54,60	10,12	52,43	71,55
		AOSnPB	95,40	30,67	49,86	72,68
	bezpośrednia	ANP	13,20	1,00	40,28	79,22
		AOSnPB	86,80	28,28	50,76	71,58

ŹRÓDŁO: opracowanie własne.

SOURCE: own elaboration.

Symulacje bitew w ustawieniu II dostarczyły ciekawych rezultatów. W przypadku algorytmu AOSnPB ponownie otrzymano najwyższe wyniki: wszystkie gry zostały wygrane przy jego użyciu przy zachowaniu prawie 50% oddziałów zarówno w starciu z armiami używającymi algorytmu losowego, jak i w bezpośredniej walce. Zaskakującym wynikiem przy tym ustawieniu były wszystkie przegrane przez armie wykorzystujące algorytm ANP oraz ich całkowita utrata oddziałów. Zauważyć należy, że pomimo przegranych algorytm ANP charakteryzował się lepszą optymalizacją

wykorzystanych zasobów niż algorytm AOSnPB. Taki przypadek mógł być spowodowany faktem, że najbliższymi przeciwnikami były bardzo wytrzymałe oddziały, których nie udało się pokonać ze względu na zbyt małą siłę atakujących.

Ustawienie III jest przykładem typowego ustawienia jednostek w grze strategicznej. Tym razem również wykazano, że w przypadku algorytmu AOSnPB uzyskano lepsze wyniki. Różnice pomiędzy algorytmem ANP oraz AOSnPB w grze przeciw ruchom wybieranym przez algorytm losowy wyniosły 20,2% w średniej liczbie wygranych gier na korzyść algorytmu AOSnPB. W starciu bezpośrednim algorytm ANP wygrał jedynie w 3,40% przeprowadzonych bitew.

Kolejnym typowym przykładem ustawienia w grze strategicznej jest ustawienie IV. Różnica w procencie wygranych gier algorytmu AOSnPB w porównaniu do ANP wyniosła 40.8%. Procent zachowanej armii również różnił się na korzyść AOSnPB o 20.55%, przy konieczności wykonania średnio ok. 50 ruchów przez oba algorytmy. Różnica w optymalności ataków pozostawała na poziomie 1%. Jeszcze większe różnice w wynikach można zauważyć w przypadku bezpośrednich starć armii wykorzystujących algorytmy ANP oraz AOSnPB.

Eksperyment 2

Eksperyment drugi dotyczy walki z przeciwnikiem w sytuacji, gdy armie nie mają równoważnych sił. Dotyczy to ustawień V i VI. Jeśli przeciwnik dysponuje słabszą armią, to celem w walce może być nie tylko być wygrana, ale równoważnym celem może stać się odniesienie jak najmniejszych strat w armii. Jeżeli przeciwnik dysponuje silniejszą armią, poza próbą wygrania mimo wszystko, celem może stać się spowodowanie jak największych strat w armii przeciwnej. Aspekty te również były również badane w pracy. Wyniki zostały zaprezentowane w dwóch tabelach. W tabeli 2 znajdują się wyniki porównania algorytmów ANP przeciw AL oraz AOSnPB przeciw AL, natomiast w tabeli 3 umieszczone zostały wyniki bezpośredniego porównania walk armii wybierającej ruch na podstawie ANP oraz AOSnPB.

Analizując siły armii w ustawieniu V, gracz, którego armia rozmieszczona jest po lewej stronie planszy, ma około 30% więcej sił – głównie dzięki dysponowaniu większą liczbą oddziałów. Ustawienie VI jest ustawieniem symetrycznym, zaś różnica polega na tym, że gracz z armią po lewej stronie planszy ma każdego oddziału o jeden więcej w porównaniu z przeciwnikiem.

W przypadku rozgrywek ze słabszym przeciwnikiem (tab. 2), wykorzystującym algorytm losowy do wyboru ruchu, oba algorytmy (ANP, AOSnPB) wygrały wszystkie bitwy, co nie jest zaskakujące. Dużą różnicę można natomiast zobaczyć w stanie armii po walce: dla algorytmu AOSnPB procent zachowanej armii przy ustawieniu V wyniósł 63,24% i był o około 9% większy niż dla algorytmu ANP. Różnica jest jeszcze większa przy ustawieniu VI i wynosi około 20%. Algorytm AOSnPB pokonał ponad trzy razy więcej armii przeciwnika niż stracił. Ponadto wykorzystanie algorytmu AOSnPB umożliwiło pokonanie przeciwnika przy wykonaniu mniejszej liczby ruchów oraz optymalne wykorzystanie zasobów.

TAB. 2. Wyniki przeprowadzonych potyczek przy dwóch wybranych ustawieniach przeciw algorytmowi losowemu AL

TAB. 2. Results of the battles for six selected army units arrangements against AL random algorithm

	Ustawienie		P_w [%]	P_a [%]	N_r	O_a [%]
Słabszy przeciwnik	V	ANP	100,00	54,35	54,39	70,00
		AOSnPB	100,00	63,24	48,37	72,39
	VI	ANP	100,00	58,96	42,61	75,10
		AOSnPB	100,00	78,95	33,75	79,83
Silniejszy przeciwnik	V	ANP	0,20	0,03	27,57	73,49
		AOSnPB	33,60	4,57	37,09	74,27
	VI	ANP	0,02	0,02	27,82	90,22
		AOSnPB	14,80	2,33	50,57	80,27

ŹRÓDŁO: opracowanie własne.

SOURCE: own elaboration.

Analizując wyniki przy rozgrywkach z silniejszym przeciwnikiem (tab. 2), zamieniono armie stronami, czyli przeciwnik dysponował liczniejszą o około 30% armią. Należy zwrócić uwagę na to, że pomimo dużej dysproporcji w siłach armii wykorzystanie algorytmu AOSnPB do wybierania ruchu pozwoliło na wygranie 14,80% w starciu z silniejszym przeciwnikiem. W przypadku algorytmu ANP było to jedynie 0,02% przypadków. Ciekawym zjawiskiem jest znacznie, bo prawie dwukrotnie, większa liczba ruchów potrzebnych AOSnPB w rozgrywce w porównaniu z algorytmem ANP oraz niższa o około 10% optymalność wykorzystania zasobów, wskazująca na zastosowanie zupełnie innej strategii walki w przypadku obu algorytmów.

Wyniki walki bezpośredniej armii wykorzystujących dwa algorytmy ANP oraz AOSnPB przedstawione są w tabeli 3. W przypadku przewagi armii wykorzystującej algorytm AOSnPB wszystkie bitwy zostały wygrane w obydwu analizowanych ustawieniach. Nie udało się potwierdzić tego wyniku dla algorytmu ANP: w ustawieniu V wygranych zostało 69% starć, a w VI 93,4%. Algorytm AOSnPB uzyskał w ustawieniu V 31% wygranych starć.

TAB. 3. Wyniki przeprowadzonych potyczek przy dwóch wybranych ustawieniach

TAB. 3. Results of the battles for six selected army units arrangements

	Ustawienie		P_w [%]	P_a [%]	N_r	O_a [%]
Przewaga armii ANP	V	ANP	69,00	12,47	60,64	72,72
		AOSnPB	31,00	4,53	40,70	74,79
	VI	ANP	93,40	30,39	45,27	76,39
		AOSnPB	6,60	0,96	53,76	83,52
Przewaga armii AOSnPB	V	ANP	0,00	32,04	18,65	75,83
		AOSnPB	100,00	60,83	45,38	72,36
	VI	ANP	0,00	0,00	20,62	91,84
		AOSnPB	100,00	77,35	33,27	80,55

ŹRÓDŁO: opracowanie własne.

SOURCE: own elaboration.

Ciekawym spostrzeżeniem jest to, że w ustawieniu V armia wykorzystująca algorytm AOSnPB wygrała około pięć razy więcej gier niż w ustawieniu VI, co mogłoby oznaczać, że lepsze działanie algorytmu AOSnPB związane jest z preferowaniem walk z większą liczbą jednostek przeciwnika.

5. Wnioski

W pracy porównane zostały trzy algorytmy strategii walki z przeciwnikiem; algorytm losowy, najbliższego przeciwnika oraz oceniający stan na polu bitwy. Wykazano przewagę algorytmu AOSnPB w porównaniu z pozostałymi dwoma algorytmami oraz oszacowano wartość tej przewagi na podstawie takich statystyk jak procent wygranych gier, liczba ruchów, ale również oceniając stan armii: procent zachowanej armii oraz optymalność wykorzystania zasobów. Na podstawie eksperymentów wykazano także przewagę algorytmu ANP nad algorytmem losowym.

Zaprezentowany algorytm oceny stanu na polu bitwy może być dalej rozwijany w kilku możliwych kierunkach. Pierwszym z nich jest analizowanie nie jednego, ale całego ciągu ruchów gracza. Wymaga to jednak dużych zasobów obliczeniowych. Innym pomysłem jest dalsza praca nad algorytmem dobierania wag w funkcji oceny stanu na polu walki w celu znalezienia rozwiązania optymalnego.

Literatura

1. Botwinnik M. *Computers in Chess*. Springer-Verlag; 1984.
2. Dąbrowski A. *Porównanie strategii walki z przeciwnikiem na polu bitwy w turowej grze strategicznej*, praca magisterska; 2018.
3. Dąbrowski A, Malesza S. *Strategiczna gra rozgrywana w systemie tur z elementami RPG*, praca inżynierska; 2017.
4. Goldberg DE, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Publ. Comp., Inc.; 1989.
5. Pawlewicz J. *Techniki sztucznej inteligencji w programach grających*; 2010. Dostępny w Internecie: <https://www.mimuw.edu.pl/~pan/gry.pdf>.
6. *Słownik Gracza*. Dostępny w Internecie: <https://www.gry-online.pl>.
7. Tadeusiewicz R. *Sieci neuronowe*. Warszawa: Akademicka Oficyna Wydawnicza RM; 1993.
8. *Unreal Engine 4 Documentation*. Dostępny w Internecie: <https://docs.unrealengine.com/en-us>.

Streszczenie

W pracy przedstawiono porównanie wybranych trzech algorytmów wyboru ruchu w trakcie rozgrywki w strategicznej grze turowej: algorytm losowy (AL), algorytm najbliższego przeciwnika (ANP) oraz algorytm oceniający stan na polu bitwy (AOSnPB). W celu przeprowadzenia eksperymentów wybrano sześć ustawień obu armii na planszy, przy czym cztery pierwsze ustawienia dotyczyły przypadku zrównoważonych sił w obu armiach, natomiast dwa kolejne dotyczyły gry z silniejszym i ze słabszym przeciwnikiem. Algorytmy ANP oraz AOSnPB porównywano z algorytmem losowym jako punktem odniesienia oraz pomiędzy sobą, by oszacować zarówno prawdopodobieństwo wygranej, jak i bardziej szczegółowe statystyki, np. stan armii po bitwie. Jest to szczególnie ważne w przypadku braku zrównoważenia sił po obu stronach. Wykazano, że algorytm oceniający stan na polu bitwy daje znacząco lepsze wyniki w porównaniu z pozostałymi algorytmami oraz oszacowano te różnice.

Słowa kluczowe: gra turowa, gra strategiczna, algorytm

Summary

Selected battles strategies in turn-based strategy game

The paper presents a comparison of three selected algorithms of motion selection during the game in a turn-based strategic game: random algorithm (AL), nearest opponent (ANP) and assessing the state on the battlefield (AOSnPB). In order to conduct experiments, six settings of both armies were chosen on the board, the first four concerned the case of balanced forces in both armies, while the two concerned games with a stronger and a weaker opponent. The ANP and AOSnPB algorithms were compared with a random algorithm as a reference and among themselves to estimate both the probability of winning and more detailed statistics, such as the state of the army after

the battle. This is especially important if the balance of forces on both sides is violated. It has been shown that the algorithm assessing the state on the battlefield gives significantly better results compared to other algorithms, and the differences were estimated.

Keywords: turn-based game, strategy game, algorithm