



Interaktywny symulator sytuacji powietrznej

MICHAŁ MILLER-KOPYT¹, ANDRZEJ WITCZAK²

Wojskowa Akademia Techniczna, ¹Wydział Mechatroniki, Uzbrojenia i Lotnictwa,
²Wydział Elektroniki, ul. gen. S. Kaliskiego 2, 00-908 Warszawa,
michal.millerkopyt@gmail.com, andrzej.witczak@wat.edu.pl

Streszczenie. Symulatory pełnią obecnie istotną rolę w szkoleniu i trenowaniu personelu wojskowego. Pozwalają między innymi na ćwiczenie trudnych bądź niemożliwych do wykonania w rzeczywistości scenariuszy bojowych. Technologie symulacji wirtualnej rzeczywistości, jak również sposoby ich współdziałania między sobą oraz obiektami rzeczywistymi są ciągle rozwijane. Artykuł porusza zagadnienie interaktywnych rozproszonych symulacji, przedstawiając ich rozwój technologiczny oraz cele, jakie przyświecały ich autorom. Po analizie metod interaktywnych symulacji wirtualnej rzeczywistości sformułowano założenia projektu oraz interakcje zachodzące pomiędzy obiektami interaktywnego symulatora sytuacji powietrznej. Na końcu podjęto próbę stworzenia interaktywnego symulatora sytuacji powietrznej zgodnie ze sformułowanymi założeniami. W rezultacie udało się zaprogramować symulator pozwalający na badanie interakcji pomiędzy systemami ostrzegania radiolokacyjnego oraz pierwotnymi radarami naziemnymi. Nie udało się zrealizować wszystkich założeń, między innymi pełnej funkcjonalności ze standardem DIS, jednak projekt ten może być dalej rozwijany, pozwalając na wprowadzenie całkowitej kompatybilności z tym standardem. Opracowanie symulatora jest procesem wymagającym wielu założeń, ale także kompromisów.

Słowa kluczowe: inżynieria lotnicza, symulatory, radary, rozproszona interaktywna symulacja
DOI: 10.5604/01.3001.0016.2812

1. Wprowadzenie

Podstawą skutecznego działania współczesnych wojsk jest użytkowanie odpowiednio skutecznej technologii oraz doskonała umiejętność jej efektywnego wykorzystania, uzyskana w rezultacie właściwie zorganizowanego procesu szkolenia. O ile rozwój technologii wojskowej znacznie przyspieszył po drugiej wojnie światowej, to szkolenie żołnierzy mających wykorzystywać nowy sprzęt odbywało się na prototypach, finalnych

produktach bądź bardzo drogich symulatorach o mocno ograniczonych możliwościach. W lotnictwie symulatory były używane, by pomóc pilotom w zapoznaniu się z maszyną i podstawowymi procedurami jej wykorzystania zgodnie z predeterminowanymi scenariuszami. Prowadziło to do uzyskiwania i doskonalenia głównie umiejętności posługiwania się danym systemem uzbrojenia — bez możliwości szkolenia w taki sposób zdolności współdziałania wielu jednostek należących do różnych rodzajów wojsk. Symultaniczna praca była bardzo ograniczona i w większości przypadków niemożliwa, a jeżeli próbowano ją uzyskać, to jedynie drogą podłączenia symulatorów bezpośrednio ze sobą [1]. Podążanie za rosnącymi potrzebami szkoleniowymi żołnierzy oraz tworzeniem nowych strategii działań w przypadku różnych możliwych scenariuszy potencjalnego pola walki, w których uzyskiwano by zarówno akceptowalną szansę przeżycia, jak i zwycięstwa, wymagało opracowania coraz bardziej zaawansowanych metod symulacji różnych scenariuszy pola walki. Szybko okazało się, że nieopłacalne i logistycznie ciężkie do zorganizowania jest fizyczne odgrywanie scenariuszy bojowych, jeżeli bierze w nich udział dużo jednostek.

Zaczęto zatem pracować nad technologią pozwalającą na przeprowadzanie działań bojowych wirtualnie, we wspólnym środowisku, mimo nieobecności wszystkich członków symulacji w tym samym symulatorze, pomieszczeniu czy nawet kraju. Dążono do umożliwienia wojskom stacjonującym w różnych krajach — czy armiom różnych krajów — przeprowadzania wspólnych manewrów wojskowych bez potrzeby eksploatacji sprzętu wojskowego oraz w miejscach, które nie są dostępne w świecie rzeczywistym. W domenie lotniczej pojawiło się pojęcie interaktywnego symulatora sytuacji powietrznej.

2. Technologie interaktywnych rozproszonych symulacji

Powstałe symulatory ograniczone były jak dotąd do współdziałania tylko między takimi samymi wersjami oraz w obrębie sieci wewnętrznej LAN. Stanowiło to problem, ponieważ znacząco ograniczało funkcjonalność takich symulatorów w przeprowadzaniu ćwiczeń na większą skalę. Prace nad standaryzacją współdziałania rozproszonych interaktywnych symulacji miały rozwiązać ten problem. Powstało kilka standardów mających na celu umożliwienie współpracy nie tylko pomiędzy różnymi symulatorami tego samego typu, lecz także między różnymi typami symulacji.

2.1. SIMulation NETworking

Na początku lat 80. XX wieku amerykańska Agencja Zaawansowanych Projektów Badawczych w Obszarze Obronności — DARPA (ang. *Defense Advanced Research Projects Agency*) zaczęła prace badawcze nad prototypem systemu sieci rozproszonych

symulatorów do działań bojowych, mającym pozwolić wojsku Stanów Zjednoczonych na przeprowadzanie wirtualnych symulacji interaktywnych scenariuszy pola walki, w których jednocześnie mogłyby brać udział jednostki zlokalizowane w różnych miejscach w kraju i na świecie. Projekt miał pokazać opłacalność i efektywność takiego rozwiązania. Od nowo budowanego systemu wymagano:

1. Możliwości wsparcia setek, a nawet tysięcy rozproszonych geograficznie jednostek;
2. Zapewnienia, aby każdy obiekt (symulacja) był heterogeniczny;
3. Możliwie niskiego kosztu budowy systemu;
4. Rozproszenia przetwarzania symulacji (brak jednostki centralnej);
5. Objęcia wymianą informacji systemów wizyjnych i radiowych;
6. Działania systemu w czasie rzeczywistym [2].

Rezultatem prac badawczych w 1984 roku [3] była sieć SIMNET (ang. *SIMulation NETworking*). Program zakończył się instalacją około 250 symulatorów w dziewięciu ośrodkach treningowych, w tym czterech w Europie i dwóch ośrodkach rozwojowych. Wojska USA użyły SIMNET-u między innymi w przygotowaniach do operacji Pustynna Burza w 1991 roku.

2.2. Aggregate Level Simulation Protocol

W styczniu 1990 roku DARPA wyszła z inicjatywą analizy procesu przeprowadzania rozproszonych gier wojennych w celu usystematyzowania i uogólnienia projektowania interfejsów symulacji konstruktywnych, używających modeli matematycznych opartych na prawach Lanchestera [4], które wykorzystywane są do przeprowadzania ćwiczeń na najwyższych szczeblach dowodzenia. DARPA zleciła to zadanie organizacji MITRE, zajmującej się zarządzaniem federalnie finansowanymi ośrodkami badań i rozwoju w Stanach Zjednoczonych. Inicjatywa ta była rezultatem między innymi bardzo dobrego odbioru sieci symulatorów SIMNET, a projekt został rozbudowany o dodatkowe założenia względem tego systemu:

1. Możliwość zarządzania czasem symulacji — czas musi być wspólny pomiędzy wszystkimi procesami biorącymi udział w symulacji;
2. System zarządzania danymi — wspólny schemat reprezentacji stanu poszczególnych symulacji;
3. Niezależność architekuralna — możliwość wspólnego działania symulatorów zbudowanych z różnymi interfejsami użytkownika i opartych na różnych językach programistycznych [5].

Architektura systemu ALSP składała się z procesu ALSP Broadcast Emulator (ABE) zajmującego się odbieraniem i przesyłaniem danych pomiędzy aktorami (symulacjami), procesów ALSP Common Module, których rolą było filtrowanie danych docierających do poszczególnych aktorów, a także tłumaczy zajmujących się przekładaniem wiadomości struktury ALSP na język zrozumiały dla poszczególnych aktorów.

2.3. Rozproszona interaktywna symulacja

Pojęcie rozproszonej interaktywnej symulacji DIS (ang. *Distributed Interactive Simulation*) odnosi się do standardu IEEE 1278, powstałego na bazie protokołów komunikacji między symulatorami sieci SIMNET, który został zatwierdzony 17 marca 1993 roku. Opracowanie standardu dotyczącego tworzenia rozproszonych interaktywnych symulacji miało na celu umożliwienie współpracy pomiędzy symulacjami tworzonymi przez różnych producentów. Wcześniejsze rozwiązania opierały się na łączeniu symulacji obiektów zagregowanych, takich jak oddziały militarne, przeznaczonych do szkoleń na wysokim szczeblu dowodzenia. Standard IEEE 1278.1-2012 jest najbardziej aktualnym dokumentem opisującym protokoły aplikacji. Definiuje on parametry pakietów danych PDU (ang. *Protocol Data Units*) przesyłanych pomiędzy instancjami symulacji oraz komputerem zarządzającym symulacją.

PDU są odpowiedzialne za przenoszenie w czasie rzeczywistym danych zawierających informacje o zdarzeniach symulacyjnych i stanowią podstawę strumienia informacji w symulacji rozproszonej zgodnej ze standardami DIS. Najnowsza wersja protokołu obejmuje 72 typy komunikatów PDU [6]. Przestrzeganie standardu przez producentów stanowisk symulacji konstruktywnych, trenażerów i rzeczywistych symulatorów oraz systemów łączących obiekty rzeczywiste z ćwiczeniem symulacyjnym gwarantuje poprawne współdziałanie tych systemów w rozproszonym systemie symulacji działań bojowych.

2.4. High Level Architecture

Standard HLA definiowany jest jako architektura frameworka systemów symulacyjnych (platform programistycznych) mająca umożliwić współdziałanie i powtórny użytek aplikacji symulacyjnych. Osiągnięte jest to przez budowanie lub modyfikowanie oprogramowania w kierunku współdziałania we wspólnym połączonym środowisku symulacyjnym nazwanym federacją. Federacja składa się z federatów, infrastruktury uruchomieniowej RTI (ang. *Runtime Infrastructure*) oraz pliku FOM (ang. *Federation Object Model*), który jest modelem opisującym, co znajduje się w obiegu wymiany danych konkretnej federacji. Za federata uznaje się każdy pojedynczy symulator lub instancję aplikacji symulacyjnej będącej członkiem federacji. Każdy federat posiada swój plik SOM (ang. *Simulation Object Model*) opisujący dane, które wysyła i odbiera z federacji. Gdy nowy federat dołącza do federacji, pobiera dane z pliku FOM, który zawiera pliki SOM obecnych już federatów i jest aktualizowany wraz z dołączaniem i odłączaniem członków federacji. Wymiana danych w federacji odbywa się przez obiekty za pośrednictwem infrastruktury uruchomieniowej RTI na podstawie modelu FOM. Standard HLA definiuje, jak FOM powinien być zbudowany poprzez szablon modelu obiektowego OMT (ang. *Object Model Template*). Każdy federat uczestniczy w wymianie

danych zgodnie z wzorcem publikuj/subskrybuj (Pub/sub), deklarując do RTI poprzez SOM, jakich danych potrzebuje i jakie udostępnia. Dane te składają się z klas obiektów i ich atrybutów oraz klas interakcji [7].

3. Projekt interaktywnego symulatora sytuacji powietrznej

Interaktywnym symulatorem sytuacji powietrznej nazywamy system umożliwiającą interakcję pomiędzy jednostkami rzeczywistymi, obiektami kontrolowanymi przez komputer w automatyczny lub predeterminowany sposób oraz jednostkami wirtualnymi kontrolowanymi przez człowieka (*human-in-the-loop*) — wszystkie obiekty biorące udział w symulacji są jednostkami powietrznymi oraz lotniczymi urządzeniami naziemnymi i wchodzi z sobą w interakcję we wspólnym symulowanym środowisku. W związku z zapotrzebowaniem na technologie symulacyjne zdecydowano się na podjęcie próby realizacji symulatora sytuacji powietrznej pozwalającego na badanie interakcji pomiędzy obiektami powietrznymi i naziemnymi [8].

3.1. Założenia projektu

Pierwszym krokiem było sformułowanie założeń tworzonego projektu:

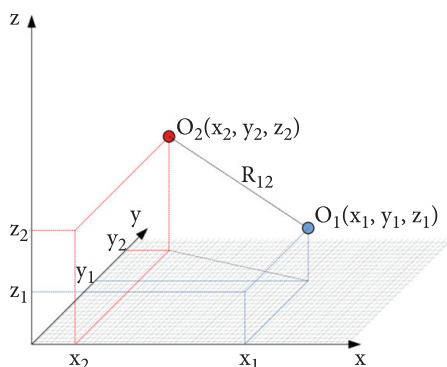
1. Umieszczenie jednostek w prostokątnym globalnym układzie współrzędnych;
2. Kulistość Ziemi jest symulowana w ograniczonym zakresie przez modelowanie horyzontu radiowego;
3. Zaprogramowanie interakcji pomiędzy statkami powietrznymi i naziemnymi radarami pierwotnymi;
4. Kompatybilność rozwiązania z protokołami standardu DIS;
5. Uwzględnienie funkcjonalności bieżącego sterowania jednostkami powietrznymi.

Na podstawie sformułowanych założeń przystąpiono do określenia warunków przebiegu interakcji pomiędzy obiektami. Jeżeli dwa obiekty O_1 i O_2 znajdują się w tym samym układzie współrzędnych (rys. 1), ich odległość od siebie obliczana jest według wzoru:

$$R_{12} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (1)$$

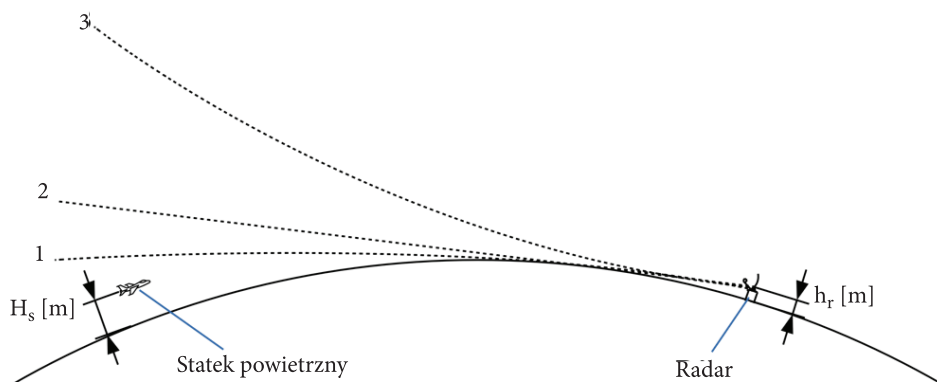
Symulacja horyzontu radiowego opiera się na wzorze określającym maksymalną odległość, w jakiej dwa obiekty mogą się widzieć ze względu na kulistość Ziemi (rys. 2).

$$R_G = 4,12 \left(\sqrt{H_s} + \sqrt{h_r} \right) \quad (2)$$



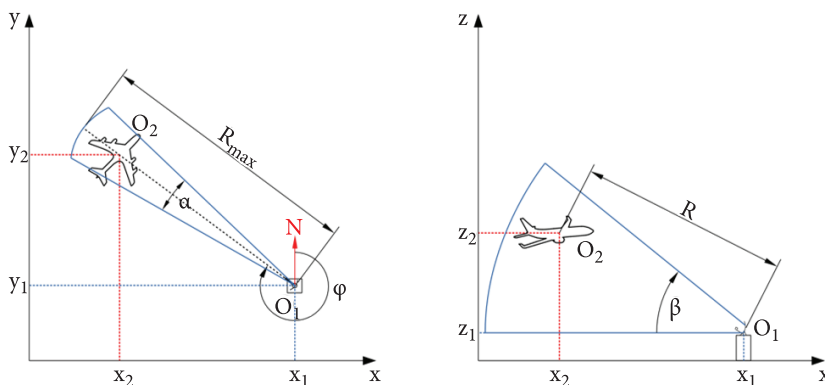
Rys. 1. Dwa obiekty umieszczone w globalnym układzie współrzędnych prostokątnych w odległości R_{12} od siebie

Użyty w symulatorze wzór uwzględnia refrakcję normalną:



Rys. 2. Zobrazowanie horyzontu radiowego: 1 — refrakcja normalna, 2 — brak refrakcji, 3 — refrakcja ujemna

Jeżeli oś y globalnego układu współrzędnych przedstawia kierunek północny, możemy stworzyć równoległą oś odniesienia zaczepioną w osi obrotu radaru naziemnego O_1 (rys. 3). Oś ta będzie wskazywała północ i służyła jako odniesienie do wyznaczania azymutu φ . Odległość wykrytego obiektu od radaru będzie oznaczona zmienną R . Zasięg maksymalny radaru wynosi R_{\max} . Pozioma szerokość kątowa wiązki radaru będzie oznaczona jako α , a pionowa szerokość kątowa wiązki jako β .



Rys. 3. Zobrazowanie dwuwymiarowe radaru i statku powietrznego znajdujących się w trójwymiarowym układzie współrzędnych kartezjańskich. Po lewej rzut na płaszczyznę xy , po prawej na płaszczyznę zx

Chcąc obliczyć azymut, na którym znajduje się obiekt O_2 względem obiektu O_1 , należy wykorzystać ich koordynaty x i y . Trzeba znaleźć różnicę w ich położeniu, zatem wprowadzono zmienną $\Delta x = x_2 - x_1$ oraz $\Delta y = y_2 - y_1$. Zależnie od wartości Δx i Δy azymut przyjmie następujące wartości:

1. $\Delta x \geq 0 \cap \Delta y > 0 \rightarrow \varphi = \arctg \left(\left| \frac{\Delta x}{\Delta y} \right| \right) [^\circ]$
2. $\Delta x > 0 \cap \Delta y = 0 \rightarrow \varphi = 90 [^\circ]$
3. $\Delta x \geq 0 \cap \Delta y < 0 \rightarrow \varphi = \arctg \left(\left| \frac{\Delta y}{\Delta x} \right| \right) + 90 [^\circ]$
4. $\Delta x \leq 0 \cap \Delta y < 0 \rightarrow \varphi = \arctg \left(\left| \frac{\Delta x}{\Delta y} \right| \right) + 180 [^\circ]$
5. $\Delta x < 0 \cap \Delta y = 0 \rightarrow \varphi = 270 [^\circ]$
6. $\Delta x < 0 \cap \Delta y > 0 \rightarrow \varphi = \arctg \left(\left| \frac{\Delta y}{\Delta x} \right| \right) + 270 [^\circ]$
7. $\Delta x = 0 \cap \Delta y = 0 \rightarrow \varphi = \text{nie istnieje}$

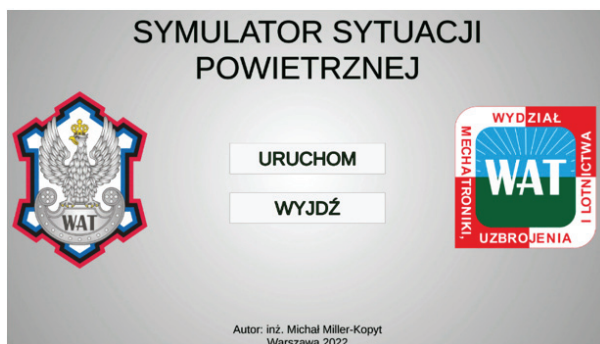
W przypadku statków powietrznych założono uwzględnienie symulacji systemu ostrzegania, który powiadamia pilota o opromieniowaniu jego statku powietrznego przez wiązkę radaru.

3.2. Narzędzia użyte podczas tworzenia projektu

Do opracowania interaktywnego symulatora sytuacji powietrznej użyto silnika do tworzenia gier Unity. Narzędzie to pozwala na całkowite opracowanie aplikacji z grafiką trójwymiarową lub dwuwymiarową od zera. Dodatkowo w celu zaprogramowania odpowiednich reakcji obiektów symulacji opracowano odpowiednie skrypty w języku C# za pomocą środowiska programistycznego Visual Studio.

3.3. Działanie aplikacji

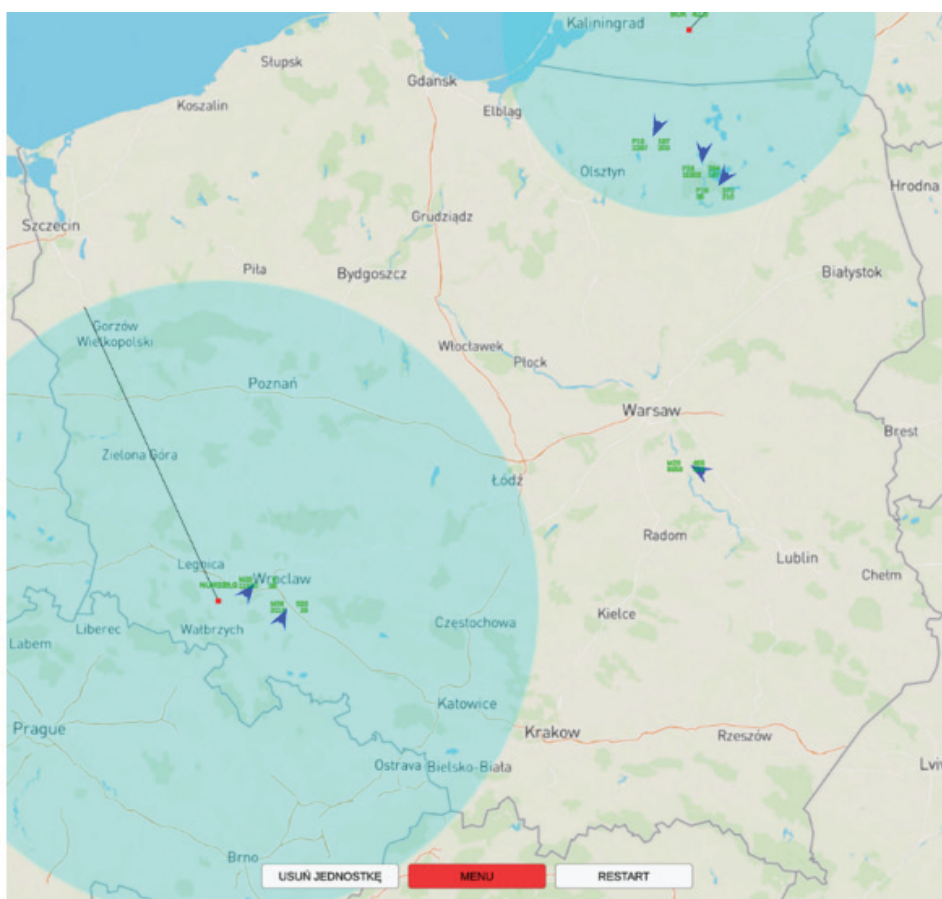
Po uruchomieniu aplikacji pojawia się jej menu główne (rys. 4).



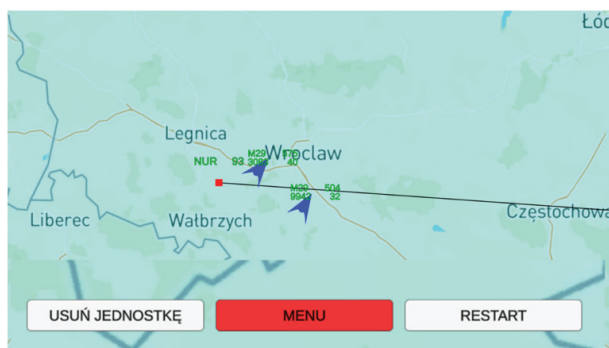
Rys. 4. Menu główne aplikacji

Po kliknięciu przycisku uruchom przed użytkownikiem pojawia się ekran główny symulacji (rys. 5), na którym widać mapę Polski i otaczających ją terenów, czyli wirtualne środowisko, w którym rozlokowano jednostki. Niebieskie strzałki symbolizują obiekty powietrzne, natomiast czerwone kwadraty reprezentują radary naziemne.

Używając klawiszy „WASD” oraz „Q” i „E”, można przybliżyć mapę i nawigować po niej (rys. 6). Zaznaczając jednostki powietrzne, można zmieniać ich wysokość klawiszami „Page Up” oraz „Page Down”. Po wybraniu jednostki można kliknąć prawym klawiszem myszki w dowolne miejsce na mapie, aby ta jednostka się tam udała. Na dole ekranu znajdują się przyciski pozwalające na usunięcie wybranej jednostki, powrót do menu lub restart symulacji.

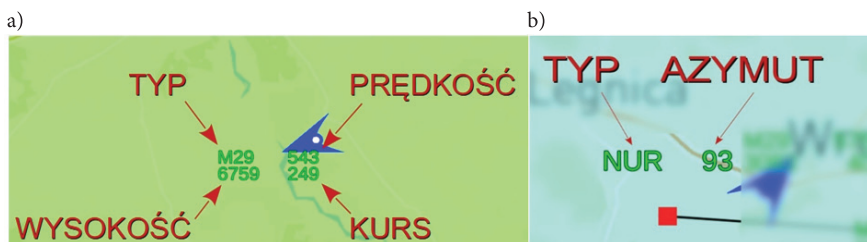


Rys. 5. Ekran główny symulacji



Rys. 6. Widok okna symulatora po przybliżeniu mapy z powiększonym paskiem menu

Radary widoczne są w aplikacji jako czerwone kwadraty. Znajdują się one w centrach niebieskich kół oznaczających ich zasięgi. Dla radaru NUR wynosi on 240 kilometrów, a dla radaru BUK 140 kilometrów. Oprócz tego obracająca się wokół środka radaru czarna linia symbolizuje obecne położenie jego wiązki radarowej. Nad symbolem jednostki znajdują się jej dane — w przypadku radaru będzie to jego typ oraz azymut obecnie obserwowany przez radar (rys. 7a).



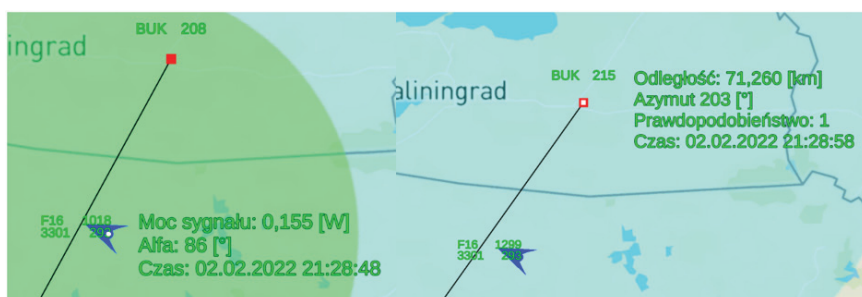
Rys. 7. Jednostka typu: a) radar (po lewej) oraz b) samolot (po prawej) wraz z jej parametrami

Oprócz radarów na mapie są także jednostki powietrzne. W ich pobliżu znajduje się zielony tekst opisujący typ jednostki, prędkość w kilometrach na godzinę, wysokość w metrach oraz kurs w stopniach (rys. 7b). Jeżeli jednostka jest zaznaczona, wokół niej pojawi się zielone koło symbolizujące zasięg jej systemu RWR. Naciskając lewy przycisk myszy i przeciągając kursorem nad jednostką, można ją zaznaczyć (rys. 8).



Rys. 8. Funkcja zaznaczania jednostek

Jeżeli zaznaczona jednostka powietrzna (biały kształt w środku ikony) wykryje sygnał radaru, spełnione są warunki horyzontu radiowego i moc radaru jest większa niż moc progowa systemu RWR, obok niej wyświetlą się takie parametry jak siła sygnału, kierunek, z którego dobiegł sygnał względem SP, oraz czas systemowy w momencie wykrycia (rys. 9a). W przypadku zaznaczonej jednostki radarowej — jeżeli spełnione są warunki widoczności — w momencie wykrycia pojawi się obok niej informacja o odległości do wykrytego obiektu, jego azymucie, prawdopodobieństwo wykrycia oraz czas systemowy w momencie wykrycia (rys. 9b).



Rys. 9. Raport wykrycia opromieniowania przez: a) system RWR (po lewej)
i b) radar naziemny (po prawej)

4. Podsumowanie

Sieć symulatorów SIMNET była pierwszym krokiem w rozwoju interaktywnych rozproszonych symulacji w zagadnieniu symulowania pojedynczych jednostek. Wraz z systemem symulacji konstruktywnych ALSP zbudowały one bazę do stworzenia obecnie używanych standardów, takich jak DIS oraz HLA. Niestety ze względu na problemy techniczne oraz ograniczenia czasowe nie udało się zaimplementować pełnej zgodności ze standardem DIS w symulatorze. Przygotowano jednak funkcjonalność przechowywania danych typu jednostek opisanych w standardzie, co pozwala na dodanie możliwości współpracy z protokołem DIS do tego projektu w przyszłości. Dane o wykryciach wyświetlane użytkownikowi można przekonwertować na odpowiednie paczki danych PDU za pomocą biblioteki OpenDIS.

Stworzony projekt pozwala na dodawanie różnych typów statków powietrznych oraz radarów w edytorze silnika Unity, w zależności od potrzeb. Dodane typy obiektów następnie mogą zostać umieszczone w scenie i poruszać się w obrębie wirtualnego środowiska, jakim jest mapa Polski. Opracowana funkcjonalność zapewnia realistyczne wykrywanie jednostek przez pierwotny radar naziemny. W przypadku jednostek powietrznych zapewniono funkcjonalność poruszania się po mapie, zmiany wysokości oraz wykrywania opromieniowania radarowego przez system RWR. Dodawanie nowych jednostek w edytorze Unity pozwala zmieniać parametry ich poruszania się, takie jak pułap, prędkość maksymalna, zasięg odborników RWR czy ich moc progowa wykrywania.

Przy dalszym rozwoju opracowany symulator sytuacji powietrznej mógłby posłużyć do szkolenia kontrolerów lotu, monitorowania przebiegu ćwiczeń wojskowych na dużą skalę czy przeprowadzania eksperymentów radarowych. Symulatory są bardzo cennym narzędziem każdej armii, a także wielu różnych instytucji, dlatego ich rozwój będzie kontynuowany przez długi czas.

Źródło finansowania pracy — środki własne autorów.

Artykuł opracowany na podstawie referatu wygłoszonego na XXXVI Konferencji Elektroniki, Telekomunikacji i Energetyki Studentów i Młodych Naukowców SECON 2022 — zorganizowanej przez Wydział Elektroniki Wojskowej Akademii Technicznej w Warszawie w dniach 26-27 kwietnia 2022 r.

Artykuł wpłynął do redakcji 7.06.2022 r. Zatwierdzono do publikacji 8.07.2022 r.

Michał Miller-Kopyt <https://orcid.org/0000-0002-0726-6552>

Andrzej Witczak <https://orcid.org/0000-0002-1290-0271>

LITERATURA

- [1] MILLER D.C., THORPE J.A., *SIMNET: the advent of simulator networking*, Proceedings of the IEEE, 83, 1995, 1114-1123.
- [2] CALVIN J., DICKENS A., GAINES B., METZGER P., MILLER D., OWEN D., *The SIMNET virtual world architecture*, Proceedings of IEEE Virtual Reality Annual International Symposium, 1993, 450-455.
- [3] National Archives Catalog, <https://catalog.archives.gov/> [dostęp: 19.10.2021].
- [4] HOFFMANN R., PROTASOWICKI T., *Modelowanie pola walki z zastosowaniem koncepcji dynamiki systemowej*, Biuletyn Instytutu Systemów Informatycznych, 12, 2013, 29-34.
- [5] WEATHERLY R.M., WILSON A.L., CANOVA B.S., PAGE E.H., ZABEK A.A., FISCHER M.C., *Advanced distributed simulation through the Aggregate Level Simulation Protocol*, „Proceedings of HICSS-29: 29th Hawaii International Conference on System Sciences”, 1996, 407-410.
- [6] PIERZCHAŁA D., SKRZYPECKI S., *Wieloagentowa i wielorozdzielcza symulacja rozproszona DisSim — VBS*, Symulacja w Badaniach i Rozwoju, 7, 1/2, 2016, 25-33.
- [7] TOPÇU O., OĞUZTÜZÜN H., *Introduction. Guide to distributed simulation with HLA*, Springer International Publishing, 2017, 12-38.
- [8] MILLER-KOPYT M., *Interaktywny symulator sytuacji powietrznej*, praca dyplomowa WAT, 2022.

M. MILLER-KOPYT, A. WITCZAK

Interactive air situation simulator

Abstract. In the present day, simulators play a significant role in training military personnel. They allow, among other things, for practicing combat scenarios that are difficult or outright impossible to execute in reality. Virtual reality technologies are being constantly developed, as are the ways of their interaction with each other and with real objects. This article brings up the topic of interactive distributed simulations, presenting their technological developments as well as the goals of their authors. After the analysis of selected methods of interactive distributed simulations, project assumptions are formulated and interactions occurring between objects in virtual airspace are stated. The last part of the article describes the attempt of creating an interactive air situation simulator in accordance with the formulated assumptions. As a result of the work conducted, it was possible to program a simulator that allows us to study the interactions between radar warning receivers and ground based primary surveillance radars. Not all assumptions have been achieved, most notably full functionality with the DIS standard, however, the project can be further developed, allowing for full compatibility with this standard. Developing a simulator is a process that requires many assumptions, but also trade-offs. With correct assumptions and a standardised way of information exchange between simulators, it is possible to make a web of interoperable simulators.

Keywords: aerospace engineering, simulators, radars, distributed interactive simulation

DOI: 10.5604/01.3001.0016.2812