

## SZTUCZNA INTELIGENCJA W MODELOWANIU OBIEKTÓW LATAJĄCYCH, CZYLI MODELOWANIE ZWINNE

FRANCISZEK DUL

Wydział Mechaniczny Energetyki i Lotnictwa, Politechnika Warszawska  
e-mail: franciszek.dul@pw.edu.pl

Modelowanie obiektów latających jest niezbędne przy budowie symulatorów wszelkich rodzajów. Celem modelowania jest symulowanie ruchu obiektów w taki sposób, aby odwzorowane zachowania samolotu były akceptowalne przez pilotów. Modelowanie klasyczne, polegające na użyciu równań mechaniki klasycznej (newtonowskich praw mechaniki: twierdzeń o zmianie pędu i krętu, i innych) oraz wykorzystaniu podstawowych danych samolotu jak również wyników prób w locie okazuje się jednak niewystarczające. Tak zbudowane symulatory pilotów oceniają najczęściej jako nie spełniające wymogów szkoleniowych. Dotyczy to zwłaszcza symulatorów samolotów wysokomanewrowych oraz śmigłowców.

W pracy przedstawiono nowe podejście do modelowania obiektów latających polegające na budowie modelu hybrydowego, łączącego model klasyczny z modelami formalnymi, opartymi na sieciach neuronowych. Do budowy i dostrajania takiego modelu zastosowano pochodzącą ze sztucznej inteligencji koncepcję uczenia ze wzmocnieniem. Nowością jest oparcie uczenia na ocenach formułowanych przez pilotów, które są jakościowe, często nieprecyzyjne i zależą od pilota oraz jego aktualnego stanu psychofizycznego. Funkcje użyteczności uczenia są zatem definiowane w formie rozmytej. Wymagało to opracowania specjalnych, dedykowanych metod numerycznych, możliwych do zastosowania w wersjach interaktywnych, gdyż warunki realizacji takich projektów (działania na budowanym obiekcie, z ograniczoną dostępnością pilotów testujących, przy krótkich terminach realizacji) wykluczają możliwość użycia standardowych procedur realizowanych w trybie off-line.

Realia budowy modeli hybrydowych wpisują się w zasadnicze tezy programowania zwinnego (*agile programming*), zatem dla takiego modelowania zaproponowano przez analogię nazwę modelowania zwinnego.

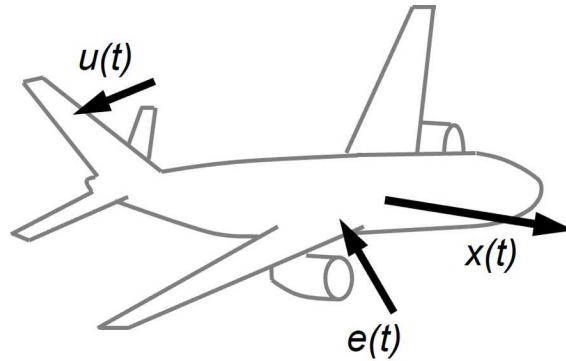
Przedstawiono wyniki zastosowania modelowania zwinnego w praktyce do budowy zaawansowanego symulatora samolotu wysokomanewrowego. Pokazano, że zastosowanie modelowania zwinnego umożliwiło uzyskanie bardzo dobrego odwzorowania własności lotnych samolotu, znacznie lepszego niż to jest możliwe przy pomocy identyfikacji modelu samolotu na podstawie prób w locie.

### 1. Wstęp

Modelowanie obiektów latających jest niezbędne przy budowie symulatorów wszelkich rodzajów [1]. Celem modelowania jest symulowanie ruchu obiektów w taki sposób, aby odwzorowane zachowania samolotu były akceptowalne przez pilotów, czyli odwzorowanie właściwych zachowań samolotu  $x(t)$  poruszającego się w określonym środowisku  $e(t)$  pod wpływem sterowania  $u(t)$  realizowanego przez pilota (rys. 1).

Kryterium „dobroci” modelu: model powinien odwzorować zachowania samolotu w sposób akceptowalny przez pilotów.

Modelowanie klasyczne, polegające na użyciu równań mechaniki klasycznej (newtonowskich praw mechaniki: twierdzeń o zmianie pędu i krętu, i innych) oraz wykorzystaniu podstawowych danych samolotu jak również wyników prób w locie, okazuje się jednak niewystarczające. Tak



Rys. 1. Model obiektu latającego

zbudowane symulatory piloci oceniają najczęściej jako nie spełniające podstawowych wymogów szkoleniowych. Dotyczy to zwłaszcza symulatorów samolotów wysokomanewrowych oraz śmigłowców.

Struktura pracy jest następująca. W kolejnych rozdziałach przedstawiono modelowanie klasyczne oraz omówiono jego wady i zalety. Następnie opisano koncepcję modelu hybrydowego. W dalszej części przedstawiono koncepcję uczenia ze wzmocnieniem modelu hybrydowego oraz omówiono jego implementację z wykorzystaniem ocen ilościowych i jakościowych formułowanych przez pilotów. Skuteczność modelowania hybrydowego przedstawiono na przykładzie budowy modelu samolotu wysokomanewrowego.

## 2. Modelowanie klasyczne

Model matematyczny samolotu jest bardzo złożony, gdyż musi opisać wiele elementów rzeczywistego samolotu. Złożony jest z modeli płatownia, aerodynamiki, napędu, podwozia, instalacji, układu sterowania, układów nawigacyjnych i innych.

Model obiektu latającego ma postać ogólną

$$\dot{x} = f(x, u, e; \theta) \quad (2.1)$$

gdzie  $x(t)$  jest wektorem stanu obiektu,  $u(t)$  – wektorem sterowania,  $e(t)$  – wektorem zmiennych środowiska, zaś  $\theta$  jest zbiorem parametrów modelu. Operator  $f(\cdot)$  jest ogólna nieliniową funkcją zawierającą modele szczegółowe: ruchu samolotu, aerodynamiki, napędu, podwozia,...

$$f = f(f_s, f_a, f_n, f_g, \dots) \quad (2.2)$$

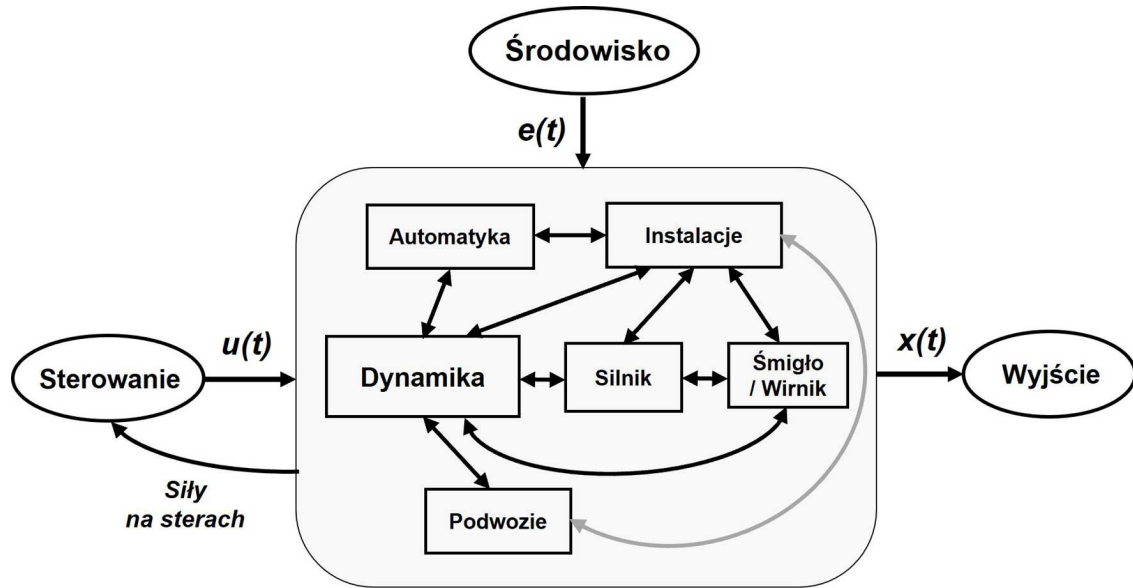
Modele podsystemów są ze sobą powiązane w sposób ogólny, na ogół nie addytywnie (rys. 2).

Każdy z modeli szczegółowych zależy od stanu  $x$ , sterowania  $u$ , środowiska  $e$ , lecz ma odrębny zbiór parametrów  $\theta_k$  (są to np. pochodne aerodynamiczne)

$$f_k = f_k(x, u, e; \theta_k) \quad (2.3)$$

Model samolotu musi być na ogół dostrojony. Polega to na dobraniu wartości parametrów w taki sposób, aby zachowania samolotu były poprawne. Poprawność oznacza zgodność stanów generowanych przez model samolotu ze stanami lotu samolotu rzeczywistego. Zgodność taką weryfikuje się, porównując stany generowane przez model ze stanami zarejestrowanymi w trakcie prób samolotu w locie lub też kierując się uwagami pilotów testujących symulator.

Model klasyczny samolotu zapewnia na ogół dość dobre odwzorowanie własności samolotu rzeczywistego, co wyraża się pozytywnymi ocenami pilotów. Dzieje się tak jednak tylko wtedy, gdy samolot posiada specyficzne cechy:



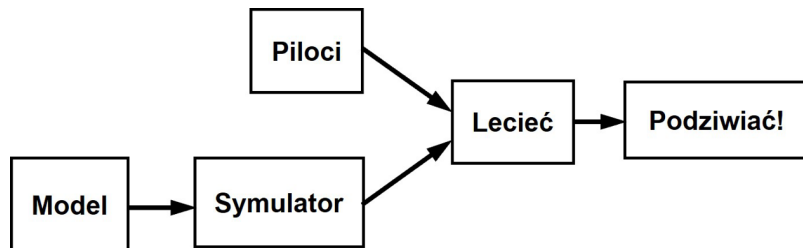
Rys. 2. Model klasyczny obiektu latającego

- jest „latającą taksówką”: wykonuje gładkie loty, niewielkie manewry,
- posiada napęd odrzutowy (a nie śmigłowy),
- posiada systemy AFCS (Automatic Flight Control System) poprawiające własności lotne samolotu,

a ponadto

- piloci nie są zbyt drobiazgowi w swoich ocenach.

W takich przypadkach dostrojenie modelu samolotu jest stosunkowo proste (rys. 3). Można tego dokonać nawet bez wykorzystania danych z prób w locie.



Rys. 3. Modelowanie klasyczne obiektu latającego dla symulatora

Modelowanie tego typu można nazwać modelowaniem w przód, przez analogię do sterowania w przód, gdyż sprzężenie zwrotne w trakcie takiego modelowania jest słabe:

- nie zmienia się struktury modelu  $f$ ,
- dostraja się parametry modelu  $\theta$ .

Modelowanie klasyczne może być oparte na ogólnych uwagach pilotów: zmienia się wybrane parametry fizyczne (np. pochodne aerodynamiczne) tak, aby w ich ocenie samolot latał „lepiej”. Nie są do tego potrzebne bardziej szczegółowe dane, np. z prób w locie.

Modele opracowane w taki sposób są akceptowane przez pilotów, jednakże tylko wtedy, gdy ich oceny nie są zbyt drobiazgowe.

### 3. Modelowanie na bazie identyfikacji parametrycznej

Bardziej dokładne modele mogą być opracowane na podstawie danych z prób w locie. Model w dalszym ciągu jest klasyczny, zaś dane z prób w locie są wykorzystywane do dostrajania modelu. Ponieważ dane takie są dużo bardziej precyzyjne niż subiektywne oceny pilotów, więc można spodziewać się znacznie lepszych własności tak poprawionego modelu samolotu.

Wykorzystanie danych z prób w locie w sposób systematyczny dokonuje się w procesie identyfikacji modelu [2]. Identyfikacja prowadzi do wyznaczenia parametrów modelu zapewniających optymalizację (minimalizację) błędów stanu i sterowania względem danych z prób w locie

$$\theta_{opt} = \arg \min_{\theta \in \Theta} J(\theta) \quad (3.1)$$

gdzie  $J(\theta)$  jest wskaźnikiem jakości

$$J(\theta) = \frac{1}{2} \sum_{k=1}^n [z(t_k) - y(t_k)]^T R^{-1} [z(t_k) - y(t_k)] \quad (3.2)$$

zaś  $z(t_k)$ ,  $y(t_k)$  są, odpowiednio, wartościami pomiarów oraz wyjść w chwilach  $t_k$ ,  $R$  jest macierzą kowariancji błędów pomiarów, a  $n$  jest liczbą pomiarów.

Identyfikacja modeli samolotów jest dziedziną bardzo dobrze rozwiniętą [2], [3], [4]. Przeprowadzona poprawnie i oparta na odpowiedniej ilości danych z prób w locie pozwala uzyskać wysoką zgodność wyjść modelu samolotu z zarejestrowanymi danymi. W przypadku danych niezaszumionych uzyskuje się bardzo dobre estymacje parametrów modelu [2].

Z identyfikacją jako narzędziem modelowania wiążą się jednak dwa problemy:

- dostępność i jakość danych,
- wrażliwość modelu na zmiany danych.

Jakość identyfikacji zależy istotnie od ilości i jakości danych. W przypadkach gdy dane z prób w locie są wysokiej jakości: są mało zaszumione oraz jest ich dużo, identyfikacja daje bardzo dobre wyniki. Niestety, w praktyce są to rzadkie przypadki. Należy mieć na uwadze, że:

- próby w locie są drogie, więc zwykle jest ich (za) mało,
- ze względów finansowo-organizacyjnych nie można na ogół uzupełniać prób „na żądanie”,
- próby w locie nie pokrywają na ogół całego zakresu stanów lotu samolotu, więc nie da się z nich „wyłuskać” niektórych istotnych informacji,
- dane z prób w locie są bardzo często (na ogół) mocno zaszumione, zwłaszcza wtedy, gdy zostały przeprowadzone latem, w burzliwej atmosferze.

Mała ilość danych oraz ich słaba jakość znacznie pogarszają jakość dostrojonego modelu.

Drugim problemem związanym z identyfikacją jest to, że model może być wrażliwy na małe zmiany wielkości wejściowych: środowiskowych  $e(t)$  lub sterowań  $u(t)$ . Oznacza to, że małe zmiany sterowania mogą powodować dużo większe zmiany stanu  $x(t)$ , a więc także wyjść modelu  $y(t)$ . Ponieważ identyfikacja parametryczna, będąca w istocie optymalizacją, cechuje się dużą wrażliwością na zmiany danych, to niewielkim różnicom wartości funkcji optymalizowanej  $J(\theta)$  mogą odpowiadać bardzo odległe wartości identyfikowanych parametrów  $\theta$ . Jednak ta cecha nie jest dużą wadą identyfikacji traktowanej jako narzędzie modelowania, gdyż wartości parametrów nie stanowią kryterium oceny modelu przez pilotów. Problemem jest to, że identyfikacja została przeprowadzona dla zarejestrowanych warunków środowiska  $e_r(t)$  oraz sterowań  $u_r(t)$ , lecz piloci testujący symulator sterują zawsze trochę inaczej niż miało to miejsce w locie testowym

$$\|u(t) - u_r(t)\| < \varepsilon \quad (3.3)$$

Wrażliwość modelu sprawia, że tak niewielka zmiana sterowań może powodować dość znaczne zmiany w zachowaniu się modelu samolotu i odchylenia jego stanu od stanu zarejestrowanego mogą być dużo większe

$$\|x(t) - x_r(t)\| \gg \varepsilon \quad (3.4)$$

co zostanie zauważone przez pilotów i ocenione negatywnie. Piloci oceniają takie zachowanie samolotu w symulatorze jako zupełnie nieakceptowalne. W rezultacie odidentyfikowany model musi być „doidentyfikowany” według procedury opartej na ocenach pilotów. Nie jest to wprawdzie wada identyfikacji, lecz modelu, ale oznacza to, że identyfikacja nie jest najlepszym sposobem modelowania.

Spektakularny przykład: problemy wielkiej firmy produkującej symulatory z symulatorem samolotu komunikacyjnego [5]. Samolot-taksówka (łatwy do modelowania), próby w locie dostarczone przez wielkiego producenta samolotu, jednakże mimo spełnienia wymagań QTG (Qualification Test Guide), dopiero udział pilota w strojeniu modelu doprowadził (po długich pracach) do zaakceptowania symulatora.

#### 4. Modelowanie neuronowe z uczeniem

Alternatywą dla modelowania klasycznego oraz identyfikacji jest modelowanie oparte na nauczanych sieciach neuronowych [6], [7]. Zamiast modelu newtonowskiego można użyć odpowiednio złożoną sieć neuronową i nauczyć ją latania. Takie podejście sprawdza się z powodzeniem w praktyce w wielu zagadnieniach sterowania w lotnictwie i nie tylko. Przykładem jest np. modelowanie dynamiki śmigłowca [8]. Do reprezentacji strategii sterowania śmigłowcem w manewrach użyto sieci neuronowej. Nauczona sieć sterowała śmigłowcem dużo bardziej precyzyjnie niż pilot. Uczenie pozwoliło wyznaczyć strategię sterowania śmigłowcem dla manewrów cechujących się najwyższym poziomem trudności.

Należy jednak zwrócić uwagę, że we wszystkich przypadkach modele neuronowe były uczone właściwego odwzorowania pojedynczych manewrów, prostych do opisanego w sposób analityczny. Nauczenie sieci neuronowej (nawet głębokiej) wszystkich zachowań samolotu wydaje się jednak problematyczne. Podejście takie wymaga przykładów uczących, którymi mogą być jedynie dane z prób w locie. Z powodów omówionych wcześniej dane takie są zazwyczaj niewystarczające. Brak dostatecznej ilości danych uczących jest powszechnym problemem uczenia maszynowego [9].

Drugim problemem podejścia opartego na uczeniu maszynowym jest wybór struktury sieci:

- zbyt prosta nie będzie w stanie nauczyć się wszystkiego,
- odpowiednio złożona, np. wielowarstwowa sieć głęboka, wymaga długiego uczenia i ogromnej ilości danych uczących.

Problemem następnym jest „niefizyczność” sieci. Sieci są strukturami formalnymi. W przeciwieństwie do modeli newtonowskich nie mają struktury jawnie wyrażającej informacje fizyczne. Własności fizyczne są zawarte niejawnie w parametrach sieci – są „rozsmarowane” po całej sieci. Uczenie zmienia zaś wszystkie parametry sieci, nie można więc dostrajać poszczególnych własności fizycznych niezależnie [10].

Uczenie ogólnej formalnej sieci neuronowej

- wymaga wielkich ilości danych,
- jest niezwykle drogie,

i dlatego niezbyt nadaje się do modelowania w warunkach rzeczywistych, w których:

- nie ma czasu na prowadzenie długotrwałych obliczeń,
- nie ma wystarczającej ilości dobrych danych uczących.

## 5. Modele hybrydowe obiektów latających

Ograniczenia modelowania klasycznego doprowadziły do sformułowania nowego paradygmatu modelowania.

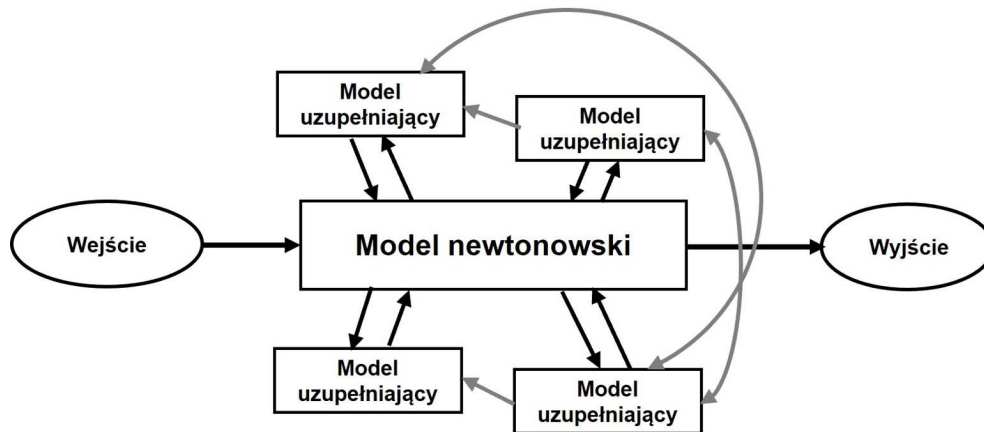
*Celem modelowania jest budowa modelu najbardziej zgodnego z odczuciami pilotów, nie zaś nie najlepszego modelu matematycznego samolotu.*

Najważniejsze cechy takiego modelowania:

- model powinien być nauczony właściwych zachowań w różnych sytuacjach,
- model nie musi być oparty na newtonowskich modelach fizycznych,
- ale model klasyczny „załatwia” 50-70% modelowania ruchu, więc na pewno warto go użyć,
- model powinien być „minimalny” – nie ma sensu uwzględnianie efektów nie obserwowanych przez pilotów.

Takie podejście doprowadziło do sformułowania koncepcji *modelu hybrydowego*.

Model hybrydowy złożony jest z modelu newtonowskiego oraz modeli formalnych (rys. 4).



Rys. 4. Model hybrydowy

Osnowę stanowi model newtonowski, fizyczny. Modele uzupełniające mają postać „łat” „doszytych” do modelu podstawowego. Modele uzupełniające mogą być formalne lub fizyczne, mogą być dowolnie powiązane ze sobą. Model hybrydowy ma postać

$$\dot{x} = f(f_N(\theta_N), f_1(\theta_1), \dots, f_n(\theta_n); x, e, u) \quad (5.1)$$

gdzie:  $f_N$  – model newtonowski,  $f_1, \dots, f_n$  – modele uzupełniające.

Wszystkie modele zależą od zmiennych stanu  $x(t)$ , środowiska  $e(t)$  oraz sterowań  $u(t)$ . Każdy model ma odrębny zbiór parametrów  $\theta_k$ .

Istotną kwestią są powiązania modeli dodatkowych z modelem newtonowskim i między sobą. Modele dodatkowe mogą być powiązane w dowolny sposób, ale najbardziej pożądane są powiązania addytywne

$$\dot{x} = f_N(x, e, u; \theta_N) + f_1(x, e, u; \theta_1) + \dots + f_n(x, e, u; \theta_n) \quad (5.2)$$

gdź pozwalają na dostrajanie wyselekcjonowanych własności lotnych samolotu.

Własność lotna  $W$  samolotu jest to zdefiniowany przez pilotów stan lotu, manewr, przebieg sił na sterach, itp. Własności lotnej  $W$  odpowiadają:

- podzbiór stanów  $X(W)$ ,

- podzbiór sterowań  $U(W)$ ,
- warunki środowiska  $E(W)$ .

Przykład:  $W$  – wznoszenie ustalone:

- podzbiór stanów  $X(W)$ :  $\varphi, \psi, \theta = \text{const}, V = \text{const}, \dots$
- podzbiór sterowań  $U(W)$ : ustalone wartości wychyleń sterów  $\delta_e, \delta_a, \delta_r$ ,
- warunki środowiska  $E(W)$ :  $\rho(h)$ , bezwietrzna atmosfera.

Nowatorską koncepcją w modelu hybrydowym jest *separowalność modeli składowych*.

W trakcie budowy modelu hybrydowego istotne jest, aby modele uzupełniające można było dostrajać niezależnie od siebie. Jest to możliwe, gdy modele są rozseparowane, gdyż wówczas nie wpływają na siebie wzajemnie. Separowalność modeli zapewnia, że strojenie jednej własności lotnej nie wpływa znacząco na inne. Znaczenie praktyczne koncepcji separowalności modeli składowych (uzupełniających) polega na tym, że umożliwia ona radykalne obniżenie kosztu budowy i strojenia modelu pełnego.

Idea separacji modeli uzupełniających może być sformalizowana w postaci warunku ortogonalności modeli względem własności lotnych.

### Ortogonalność modeli względem własności lotnych

Dla wybranej własności lotnej  $W_k$  i odpowiadającego jej podzbioru stanów  $X(W_k)$  oraz sterowań  $U(W_k)$ , przy ustalonych warunkach środowiska  $E(W_k)$ , model własności lotnej  $W_k$  jest ortogonalny względem modeli innych własności lotnych  $W_j$ , jeżeli strojenie modelu dla własności  $W_k$  nie zmienia w istotny sposób pozostałych własności  $W_j$ ,  $j \neq k$ .

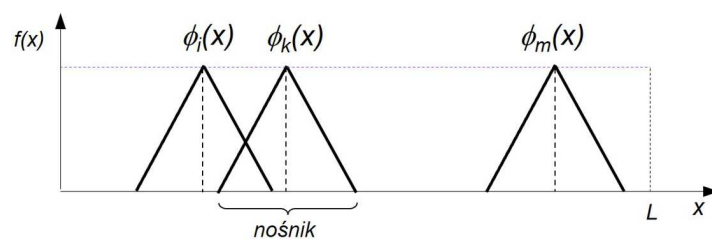
Ortogonalność względem własności lotnych można wyrazić poprzez żądanie, aby własność lotna nie zmieniała się przy zmianach parametrów innych własności lotnych

$$\frac{\partial W_j(x, e, u; \theta_j)}{\partial \theta_k} = 0 \quad j \neq k \quad (5.3)$$

Ścisła ortogonalność modeli nie jest jednak na ogół możliwa. Dlatego w praktyce sensowniejsze jest określenie koncepcji prawie-ortogonalności poprzez żądanie, aby własność lotna zmieniała się w niewielkim stopniu przy zmianach parametrów innych własności lotnych, podczas gdy zmiany parametrów tej własności lotnej powinny wywołać znaczne jej zmiany, co umożliwi jej strojenie

$$\begin{aligned} \frac{\partial W_j(x, e, u; \theta_j)}{\partial \theta_k} &< \varepsilon & j \neq k \\ \frac{\partial W_k(x, e, u; \theta_k)}{\partial \theta_k} &\gg \varepsilon \end{aligned} \quad (5.4)$$

Ortogonalność modeli względem własności lotnych jest wzorowana na pojęciu nośników funkcji bazowych w Metodzie Elementów Skończonych [11].



Rys. 5. Funkcje bazy MES – ortogonalność i prawie ortogonalność

Funkcje bazy MES są określone na podzbiorach zwartych – nośnikach. W dziedzinie  $D = [0, L]$  są prawie ortogonalne: nie zależą od siebie na dużych odległościach (np.  $\phi_k(x)$  i  $\phi_m(x)$ ), ale na bliskich „zazębiają się” ( $\phi_k(x)$  i  $\phi_i(x)$ ). Jest to ortogonalność oparta na dziedzinie funkcji, nie zaś na jej wartościach jak to ma miejsce w przypadku ortogonalności, np. funkcji harmonicznym  $\sin(k\pi x/L)$  na przedziale  $[0, L]$ .

Analogicznie, modele, które także są funkcjami, mogą być prawie ortogonalne w wielowymiarowej dziedzinie określonej jako produkt zmiennych stanu  $x$ , sterowania  $u$  oraz środowiska  $e$ ,  $D = X \times U \times E$ .

### Przykład

Modelowanie korkociągu rozwiniętego na dużych kątach natarcia nie wpływa na modelowanie lotu samolotu na podkrytycznych kątach natarcia. Występuje pełna ortogonalność korkociągu do pozostałych własności lotnych względem kąta natarcia (a więc składowych prędkości  $u$  i  $w$ )

$$D_{kork} = X_{nadkr} \times U \times E \quad X_{nadkr} : \arctg(w/u) > \alpha_{kr}$$

Modelowanie przeciągnięcia jest tylko częściowo ortogonalne, gdyż obejmuje zarówno pod- jak i nadkrytyczne kąty natarcia, więc „zahacza” ono o inne własności lotne, np. wznoszenie, podejście do lądowania

$$D_{przec} = X_{przec} \times U \times E \quad X_{przec} : \arctg(w/u) > \alpha_{kr} - 5^\circ$$

Prawie-ortogonalność modeli jest zatem abstrakcyjnym uogólnieniem koncepcji nośników zwartych bazy w MES.

Ortogonalność modeli względem własności lotnych nie jest własnością konstruktywną – nie mówi, jaka powinna być struktura modelu konkretnej własności lotnej. Daje jednak wskazówki dotyczące podzbioru dziedziny na której własność lotna jest określona.

Pożądane jest, aby ortogonalność lub prawie ortogonalność modeli dotyczyła wszystkich modelowanych własności lotnych. Jest to na ogół nieosiągalne, jednakże rozseparowanie nawet kilku własności lotnych radykalnie obniża koszt opracowania i strojenia modelu.

Wykorzystanie prawie ortogonalności modeli sprawia, że koszt opracowania i strojenia modelu hybrydowego jest nieporównanie mniejszy niż koszt uczenia ogólnej sieci neuronowej, zwłaszcza uczenia głębokiego.

## 6. Uczenie ze wzmocnieniem modelu hybrydowego

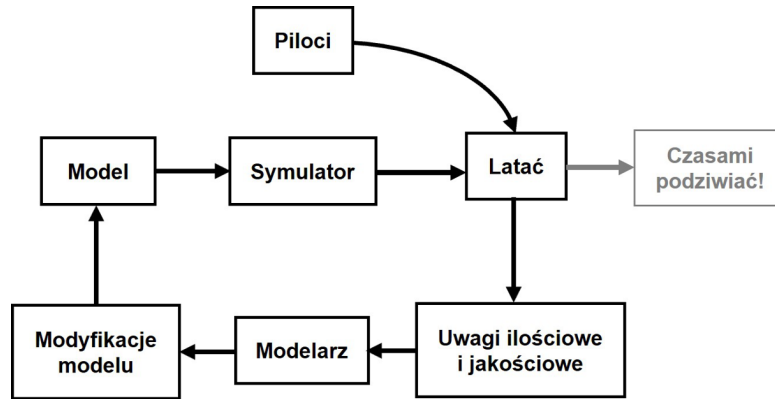
Modelowanie hybrydowe jest modelowaniem iteracyjnym z głębokim sprzężeniem zwrotnym (feedback) (rys. 6).

Sprzężenie zwrotne jest realizowane poprzez uczenie modelu. Ze względu na charakter danych (uwagi ilościowe i jakościowe formułowane przez pilotów) najlepiej do tego celu nadaje się uczenie ze wzmocnieniem (uczenie z krytykiem), Reinforcement Learning [11], [12]. Uczenie ze wzmocnieniem, w przeciwieństwie do uczenia z nauczycielem (Supervised Learning), nie wymaga danych uczących w postaci ciągu par uczących, a jednocześnie umożliwia wykorzystanie danych ilościowych, jakościowych (rozmytych), a także informacji syntetycznych (ogólnych) [11].

W uczeniu ze wzmocnieniem parametry modelu  $\theta$  wyznaczone są w oparciu o funkcję użyteczności  $Q(x, u, e; \theta)$ . Funkcja ta jest określana indywidualnie dla każdej modelowanej własności lotnej  $W$ . Uczenie modelu polega na maksymalizacji funkcji użyteczności  $Q(\theta)$  względem parametrów  $\theta$

$$\theta_{opt} = \arg \max_{\theta \in \Theta} Q(x, u, e; \theta) \quad (6.1)$$





Rys. 6. Uczenie ze wzmocnieniem modelu hybrydowego

Zależność funkcji użyteczności  $Q$  od parametrów  $\theta$  określona jest niejawnie poprzez model

$$Q(x, u, e; \theta) = Q(f(x, u, e; \theta)) \quad (6.2)$$

Dziedziną  $D$  funkcji użyteczności  $Q$  jest produkt zmiennych stanu  $x$ , sterowania  $u$  i środowiska  $e$

$$D = X \times U \times E \quad (6.3)$$

zaś obrazem (przeciwdziedziną)  $S$  mogą być różne obiekty: liczbowe (numeryczne), leksykalne, rozmyte.

Wartości funkcji użyteczności mogą mieć różne formy i poziomy precyzyjności:

— forma leksykalna, rozmyta, nieprecyzyjna

$$Q(x, u; \theta) = \{\text{za słaba siła na drążku}\} \quad (6.4)$$

— forma leksykalna, rozmyta, ilościowa

$$Q(x, u; \theta) = \{\text{zacieśnianie spirali o 10\% za szybkie}\} \quad (6.5)$$

— forma numeryczna, ilościowa

$$Q(x, u; \theta) = \{\text{prędkość wznoszenia } w = 10 \text{ m/s}\} \quad (6.6)$$

Funkcja użyteczności  $Q$  jest zatem odwzorowaniem postaci

$$Q : D \rightarrow S \quad (6.7)$$

Niestandardowość tak zdefiniowanej funkcji użyteczności polega na tym, że przekształca ona dziedzinę liczbową w zbiór wartości bardzo różnych typów.

W uczeniu ze wzmocnieniem modelu hybrydowego kluczowe jest to, że wartości funkcji użyteczności są określane subiektywnie przez pilota.

W modelowaniu hybrydowym funkcja użyteczności wykorzystywana jest w inny sposób niż to ma miejsce w klasycznym uczeniu sieci neuronowych. Uczenie oparte jest bowiem na rozwiązaniu równania nieliniowego, a nie na optymalizacji (6.1). Uczenie ze wzmocnieniem nie polega na wyznaczeniu argumentu  $\theta_{opt}$  maksymalizującego funkcję użyteczności, lecz na wyznaczeniu właściwych wartości parametrów  $\theta_{req}$  poprzez rozwiązanie równania nieliniowego

$$Q(\theta_{req}) - Q_{req} = 0 \quad (6.8)$$

w którym  $Q_{req}$  jest zadaną (żądaną) wartością funkcji użyteczności.

Wartości żądane funkcji użyteczności są określane przez pilota, a rozwiązanie ma charakter interaktywny. W kolejnych krokach:

- wykonywany jest lot testowy (w symulatorze), w którym realizuje się własność lotną,
- pilot określa wartość funkcji użyteczności  $Q$  i przekazuje ją modelarzowi,
- modelarz wprowadza ją do programu modelującego,
- program wyznacza korekcje parametrów modelu  $\Delta\theta$ ,
- korekcje parametrów są wprowadzane do oprogramowania modelu w symulatorze.

### Oszacowania funkcji użyteczności

Wartości funkcji użyteczności można interpretować na dwa sposoby:

- jako zmienną lingwistyczną,
- jako zmienną losową.

Obie interpretacje są użyteczne w modelowaniu hybrydowym.

### Funkcja użyteczności jako zmienna lingwistyczna

Oszacowania funkcji użyteczności mają charakter werbalny: pilot mówi jaką – jego zdaniem – ma ona wartość, przy czym oceny są także (najczęściej) jakościowe. Dlatego funkcję użyteczności można uznać za zmienną lingwistyczną definiowaną w teorii zbiorów rozmytych [10]. Oszacowanie wartości funkcji użyteczności można zatem interpretować jako proces wyostrażania (defuzyfikacji) zmiennej lingwistycznej. W wyniku tej operacji otrzymuje się wartość liczbową, którą można wykorzystać w algorytmie rozwiązania równania identyfikacji (6.8).

### Funkcja użyteczności jakościowa, dyskretna, trójwartościowa

Funkcja użyteczności przyjmuje najczęściej wartości nienumeryczne, jakościowe. Istotna jest przy tym monotoniczność zmian funkcji użyteczności względem zmian parametrów. W praktyce najczęstszym przypadkiem jest trójwartościowa funkcja użyteczności

$$Q = \{\text{za mało, dobrze, za dużo}\} \rightarrow \{-1, 0, 1\} \quad (6.9)$$

W tej formie można jej użyć w uogólnionym algorytmie bisekcji, idealnie nadającym się do strojenia modelu opartego na rozwiązaniu równania identyfikacji (6.8).

Algorytm bisekcji z trójwartościową funkcją użyteczności

$$\theta_1, Q_1 = Q(\theta_1) = -1$$

$$\theta_2, Q_2 = Q(\theta_2) = 1$$

Dopóki  $\theta_2 - \theta_1 > \varepsilon$  oraz  $Q_b \neq 0$ :

$$\theta_b = (\theta_1 + \theta_2)/2$$

$$Q_b = Q(\theta_b) \text{ (oszacowanie werbalne przez pilota)}$$

jeżeli  $Q_b = -1$

$$\theta_1 = \theta_b, Q_1 = Q_b$$

jeżeli  $Q_b = 1$

$$\theta_2 = \theta_b, Q_2 = Q_b$$

W praktyce zbieżność uzyskuje się w powtarzalny sposób. Stopień zawężenia  $\varepsilon$  zależy od „pływania” oszacowań funkcji użyteczności  $Q(\theta)$ . Jeżeli jednak długość przedziału  $[\theta_1, \theta_2]$  staje się niewielka, a kolejne oszacowania funkcji użyteczności nie przyjmują wartości „dobrze” (0), to oznacza to, że model cechuje się nieciągłością względem parametrów, co jest nieakceptowalne i model musi być zmodyfikowany.

**Funkcja użyteczności interpretowana jako zmienna losowa** daje możliwość jakościowego oszacowania precyzji i powtarzalności ocen pilotów. Zakłada się, że gęstość prawdopodobieństwa funkcji użyteczności  $Q(\theta)$  opisuje rozkład normalny z wartością średnią  $m_{Q(\theta)}$  i wariancją  $\sigma_{Q(\theta)}$ .

Istotne jest przy tym, aby:

- wartość średnia  $m_{Q(\theta)}$  była stabilna,
- wariancja oszacowań  $\sigma_{Q(\theta)}$  była mała.

Wartość średnia i wariancja zależą od pilotów, czasu prowadzenia prób, charakteru prób. Oszacowania pilotów są, wbrew pozorom, dość stabilne. Nawet po całodziennych nużących testach oszacowania funkcji użyteczności nie różnią się znacząco od oszacowań wcześniejszych. Różni piloci, nie zasugerowani ocenami kolegów, mają na ogół bardzo podobne odczucia. Dlatego:

- wartości średnie  $m_{Q(\theta)}$  są na ogół stabilne,
- w trakcie długotrwałych prób zmieniają się jednak wariancje oszacowań  $\sigma_{Q(\theta)}$ .

Obserwuje się jednak zjawisko rozmycia oszacowań, polegające na zmianach ocen formułowanych przez tych samych pilotów w dłuższych okresach czasu (miesiące lub lat). Dzieje się tak często dlatego, że piloci latają na różnych egzemplarzach prawdziwego samolotu (tego samego typu).

Rozmycie oszacowań zależy istotnie od dostrajanej własności lotnej:

- oszacowania zmiennych stanu: prędkości liniowych i kątowych, kątów orientacji są stosunkowo stabilne w czasie,
- dość stabilne są także oszacowania pochodnych zmian stanu (gradientów),
- oszacowania przyspieszeń liniowych i kątowych, sił na organach sterowania mogą się jednak zmieniać w dłuższych okresach czasu znacząco.

Rozmycie oszacowań pilotów można częściowo zweryfikować poprzez przeprowadzenie „ślepego testu”. Pilot informowany jest o tym, że sprawdzane są jego odczucia.

Sposoby testowania:

- zmienianie niemonotoniczne parametrów,
- wycofanie się ze zmiany,
- nie zmienianie niczego.

Nie należy jednak z tym przesadzać. Cierpliwość pilotów, nawet nastawionych pozytywnie, ma swoje granice.

## Oprogramowanie modelowania hybrydowego

Opisany sposób modelowania wykorzystujący interaktywne rozwiązywanie równania użyteczności (6.8) sprawia, że nie ma możliwości stosowania metod numerycznych w postaci standardowych algorytmów bibliotecznych. Dlatego opracowano metody numeryczne w wersji interaktywnej („ludzkiej”):

- bisekcja interaktywna,
- interaktywna metoda gradientowa.

Algorytm bisekcji interaktywnej ma postać

$$\begin{aligned} \theta_1, Q_1 &= Q(\theta_1) < Q_{req} \\ \theta_2, Q_2 &= Q(\theta_2) > Q_{req}, \\ \text{dopóki } Q_2 - Q_1 &> \varepsilon: \\ \theta_b &= (\theta_1 + \theta_2)/2 \\ Q_b &= Q(\theta_b) \text{ (ilościowe oszacowanie werbalne przez pilota)} \\ \text{jeżeli } Q_b &< Q_{req} \\ \theta_1 &= \theta_b, Q_1 = Q_b \\ \text{w przeciwnym razie} \\ \theta_2 &= \theta_b, Q_2 = Q_b \end{aligned}$$

Stopień zawężenia  $\varepsilon$  zależy od „pływania” oszacowań funkcji wartości  $Q(\theta)$ . Dziesięciokrotne zawężenie przedziału  $[Q_1, Q_2]$  wymaga na ogół przeprowadzenia trzech do pięciu prób. Należy pamiętać, że każde oszacowanie funkcji  $Q(\theta)$  to wykonanie (być może dłuższego) lotu testowego.

Gradient interaktywny werbalny umożliwia zmniejszenie liczby koniecznych testów.

Krok algorytmu gradientowego ma postać

$$\begin{aligned}\theta_1, Q_1 &= Q(\theta_1) \\ \theta_2, Q_2 &= Q(\theta_2) \\ \theta_0 &= \theta_1 + (\theta_2 - \theta_1)(Q_{req} - Q_1)/(Q_2 - Q_1) \\ Q_0 &= Q(\theta_{req}) \text{ (ilościowe oszacowanie werbalne przez pilota)} \\ \varepsilon &= Q_{req} - Q_0\end{aligned}$$

Krok gradientu wymaga przeprowadzenia tylko trzech prób. Błąd jest na ogół mały i akceptowalny często po wykonaniu tylko jednego kroku algorytmu gradientowego.

Wykorzystanie bisekcji i gradientu znacznie zmniejsza liczbę koniecznych prób. Mniejsza liczba prób zwiększa precyzję oceny pilota – jest on po prostu mniej zmęczony. Najwygodniejsza i najskuteczniejsza jest bisekcja. Użycie gradientu jest uzasadnione, gdy próba trwa długo.

### Wybór struktury modeli uczonych ze wzmocnieniem

Kwestią kluczową modelowania hybrydowego jest wybór struktury modeli uzupełniających.

Jak wiadomo, wybór struktury sieci jest piętą achillesową ogólnych metod uczenia [9], [10]. Jest to, ogólnie biorąc, sztuka modelowania. Nie można podać ścisłych zasad takiego doboru. Struktura powinna być minimalna (jak najprostsza, ale nie bardziej – A. Einstein). Jeżeli to możliwe, należy kierować się fizyką modelowanej własności lotnej: np. budując uproszczone modele opływu samolotu czy lokalnego opływu elementów samolotu w określonych warunkach lotu, budując scenariusze fizyczne awarii lub katastrof, itp.

Jeżeli podstawy fizyczne własności nie są oczywiste lub gdy model zbudowany na podstawie założeń fizycznych nie daje zadowalających rezultatów (a to się zdarza!), to pozostaje użycie sieci formalnej. W takich przypadkach należy definiować sieć jak najprostszą, z niewielką liczbą parametrów, a następnie rozbudowywać ją, jeśli zajdzie taka potrzeba.

W każdym przypadku kluczowe znaczenie ma właściwe określenie podzbiorów dziedziny (stanu, sterowań i środowiska) istotnych dla modelowanej własności lotnej, gdyż ich właściwe określenie zwiększa ortogonalność modelu do innych modeli, co znacznie ułatwia modelowanie i strojenie modelowanej własności lotnej.

### Własności uczenia ze wzmocnieniem w modelowaniu hybrydowym

W modelowaniu hybrydowym bardzo dużą rolę odgrywa czynnik ludzki. Uczenie modeli wymaga na ogół zaangażowania kilku pilotów, aby uzyskać uśrednienie ich ocen oraz zapewnić powtarzalność zachowań symulatora w późniejszej eksploatacji. Koszt modelowania hybrydowego jest bardzo wysoki, ale dzięki wykorzystaniu ortogonalności modeli jest nieporównanie niższy niż koszt uczenia ogólnej sieci neuronowej.

Proces uczenia modelu hybrydowego jest trudny do automatyzacji. Nie ma sensu opracowania algorytmów, które robiłyby to automatycznie, gdyż po prostu nie ma na to czasu, a koszt tworzenia takiego oprogramowania byłby zupełnie nieakceptowalny. Z pomocą procedur interaktywnych opisanych powyżej strojenie można przeprowadzić bardzo efektywnie.

Zalety modelowania hybrydowego:

1. Skuteczność – to działa!
2. Elastyczność – tak można modelować wszystko.
3. Niski (relatywnie) koszt procesu budowy i dostrajania modelu.

4. Pełna kontrola nad strukturą tworzonego modelu.
5. Wykorzystanie wiedzy o zjawiskach i procesach (model newtonowski).
6. Uwzględnienie informacji ilościowych i jakościowych.
7. Modelowanie w wysoce niekomfortowych warunkach rzeczywistych.

Wady modelowania hybrydowego:

1. Jest to rękodzieło szyte na miarę.
2. Proces jest trudny do sformalizowania i automatyzacji.
3. Wymagane jest bardzo intensywne zaangażowanie pilotów.
4. Kluczowe znaczenie mają przy tym intuicja i doświadczenie modelarza.

Wydaje się jednak, że zalety zdecydowanie przeważają nad wadami, czego najlepszym dowodem są sukcesy osiągnięte przy pomocy modelowania hybrydowego.

## 7. Modelowanie zwinne

Modelowanie klasyczne wraz z identyfikacją modelu na podstawie danych z prób w locie może być, przynajmniej w teorii, przeprowadzone bez udziału pilotów testujących, niejako „off-line”. Zdarza się też czasami, że symulatory z tak opracowanymi modelami zostaną akceptowane przez użytkownika. Jednakże są to sytuacje raczej wyjątkowe. Regułą bowiem jest, że piloci testujący symulator mają bardzo wiele zastrzeżeń do jego działania, w szczególności do zachowania się samolotu, czyli do jego modelu. Oznacza to, że budowanie symulatora bez ich udziału aż do końca realizacji projektu jest niezwykle ryzykowne, gdyż prawie na pewno spowoduje niezaakceptowanie symulatora w trakcie odbiorów końcowych, co skutkuje bardzo poważnymi konsekwencjami organizacyjnymi i finansowymi. Dlatego budowa symulatora, wraz z modelowaniem, muszą być prowadzone z udziałem pilotów testujących od samego początku realizacji projektu.

Powyższe uwagi dotyczą nie tylko symulatorów, lecz ogólnie realizacji projektów informatycznych. Zostało to zaobserwowane w latach 90. XX wieku, gdy liczba projektów informatycznych zaczęła wzrastać lawinowo. Opierając się na zebranych doświadczeniach, na początku XXI wieku opracowano zasady programowania zwinnego „Agile Programming” [14]. Jest to usankcjonowanie praktyki tworzenia oprogramowania w warunkach rzeczywistych, w trakcie realizacji projektu informatycznego.

Cztery podstawowe założenia programowania zwinnego zawarte w Manifeście (Agile Manifesto):

1. Ludzie i ich interakcje zamiast procedur i narzędzi.
2. Działające oprogramowanie zamiast obszernej dokumentacji.
3. Współpraca z użytkownikiem przy negocjowaniu umowy.
4. Reagowanie na potrzeby zamiast przestrzegania planu.

Podstawowe zasady programowania zwinnego można streścić następująco [14]. Najwyższy priorytet ma zadowolenie klienta, co osiąga się przede wszystkim dzięki ciągłemu wdrażaniu poprawnie działającego oprogramowania. Działające poprawnie oprogramowanie jest podstawową miarą postępu projektu. Oprogramowanie powinno być weryfikowane przez klienta jak najczęściej, najlepiej na bieżąco. Kluczową rolę pełni w tym procesie ciągła wymiana informacji poprzez bezpośrednie rozmowy wykonawcy z odbiorcą. Należy także być przygotowanym na zmiany wymagań przez klienta nawet na późnym etapie realizacji projektu. Kluczową cechą opracowywanego oprogramowania powinna być prostota, gdyż minimalizuje ona ilości koniecznej pracy oraz zmniejsza ryzyko powstawania czasami trudnych do wykrycia błędów.

Opisanie wyżej cechy programowania zwinnego mają bezpośrednio przełożenie na modelowanie realizowane w warunkach rzeczywistych. Kluczowym elementem modelowania hybrydowego

z uczeniem ze wzmocnieniem jest bowiem bardzo intensywny udział pilotów, nie tylko w sprawdzaniu poprawności działania symulatora, ale także w samym procesie modelowania.

Modelowanie hybrydowe z uczeniem ze wzmocnieniem może być więc uważane za szczególny przypadek programowania zwinnego. Dlatego można je określić jako modelowanie zwinne – Agile Modelling.

Podstawowe, szczególne cechy modelowania zwinnego można ująć następująco:

1. Oceny pilotów jako podstawa modelowania.
2. Werbalna jakościowa ewaluacja funkcji użyteczności uczenia.
3. Połączenie modeli fizycznych i formalnych.
4. Ortogonalność modeli uzupełniających.

Modelowanie zwinne wykorzystujące modelowanie hybrydowe z uczeniem ze wzmocnieniem pozwoliło na udane realizacje licznych projektów budowy symulatorów dla wielu użytkowników cywilnych i wojskowych na całym świecie.

## 8. Zastosowanie modelowania zwinnego do budowy modelu samolotu wysokomanewrowego

Modelowanie hybrydowe zostało zastosowane z powodzeniem do budowy wielu symulatorów, w tym zaawansowanego modelu samolotu wysokomanewrowego. Samolot taki, z punktu widzenia modelowania, cechuje się bardzo niekorzystnymi cechami:

- jest to wyjątkowo „narowisty rumak”, wykonujący pełną akrobację,
- posiada napęd śmigłowy o dużym nadmiarze mocy,
- nie ma zainstalowanych systemów AFCS korygujących dynamikę.

Ponadto:

- brak jest adekwatnych modeli niektórych stanów lotu i kołowania samolotu,
- liczba danych z prób w locie była duża, ale były one bardzo zaszumione,
- piloci formułowali bardzo drobiazgowo uwagi, niezwykle trudne do uwzględnienia.

Stosując modelowanie zwinne, opracowano bardzo złożony model hybrydowy tego samolotu. Modelowano kilkadziesiąt własności lotnych obejmujących pełny zakres stanów lotu i kołowania: starty, lądowania, kołowanie, precyzyjne odwzorowanie sił na sterach, loty zespołowe, pełną akrobację, precyzyjny, wielofazowy korkociąg, ślizg na ogon, a także liczne awarie, a nawet lżejsze katastrofy. Osiągnięto niezwykle dokładne (często z błędem  $< 1\%$ ) odwzorowanie zachowań samolotu we wszystkich stanach lotu. W opiniach pilotów, także nie latających na tym typie samolotu, uzyskano „bardzo dobre odwzorowanie trudnych zachowań samolotu treningowego z napędem turbinowym”.

## 9. Wnioski

Modelowanie zwinne okazało się skutecznym narzędziem tworzenia modeli w warunkach praktycznych. Najważniejszą zaletą uczenia ze wzmocnieniem modelu hybrydowego jest jego niezwykła wprost skuteczność. Podczas gdy modelowanie tradycyjne oraz modelowanie wykorzystujące identyfikację bardzo często zawodziły, modelowanie hybrydowe z uczeniem ze wzmocnieniem zawsze kończyło się sukcesem, to znaczy akceptacją symulatora przez pilotów.

Opisana metodologia modelowania udowodniła praktycznie swoją skuteczność, dając możliwość osiągnięcia wysokiej precyzji dostrojenia modeli – nawet ułamek procenta! – nieosiągalnej w inny sposób.

Kluczowym elementem modelowania hybrydowego ze wzmocnieniem jest koncepcja prawie ortogonalności modeli składowych, gdyż jej użycie pozwala radykalnie obniżyć koszt uczenia modelu.

Nowatorską koncepcją w uczeniu ze wzmocnieniem zastosowanym w modelowaniu hybrydowym stanowi interaktywne, jakościowe oszacowanie funkcji użyteczności przez pilota.

Kwestia automatyzacji procesu modelowania pozostaje na razie otwarta.

## Bibliografia

1. ALLERTON D., *Principles of Flight Simulators*, Wiley, 2009
2. JATEGAONKAR R.V., Flight vehicle system identification. A time domain methodology, *Progress in Astronautics and Aeronautics*, **216**. AIAA, Reston, VA, 2006
3. LICHOTA P., Wavelet transform-based aircraft system identification, *Journal of Guidance, Control and Dynamics*, 2023, article in press
4. LICHOTA P., SZULCZYK J., TISCHLER M.B., BERGER T., Frequency responses identification from multi-axis maneuver with simultaneous multisine inputs, *Journal of Guidance, Control and Dynamics*, **42**, 11, 2550-2556, 2019
5. PATURSKI Z., Informacje prywatne, 2021
6. HASSOUN M., *Fundamentals of Artificial Neural Networks*, The MIT Press, 1995
7. DUL F.A., Active suppression of freeplay aeroelastic vibrations of ailerons by robust control methods with incomplete measurements, *Aircraft Engineering and Aerospace Technology*, **90**, 4, 981-993, 2018
8. NG A.Y., JORDAN M.I., 2000, PEGASUS: A policy search method for large MDPs and POMDPs, *Proceedings of 16th Conf. Uncertainty in Artificial Intelligence*, 2000
9. GOODFELLOW I., BENGIO Y., COURVILLE A., *Deep Learning*, The MIT Press, Cambridge, 2016
10. FLASIŃSKI M., *Wstęp do sztucznej inteligencji*, PWN, Warszawa, 2011
11. BATHE K.J., *Finite Element Procedures*, K.J.Bathe, 2014
12. STUTTON R.S., BARTO G., *Reinforcement Learning*, 2nd Ed., The MIT Press, Cambridge, Massachusetts, 2016
13. BUSONI L., BABUSKA R., DE SCHUTTER B., ERNST D., *Reinforcement Learning and Dynamic Programming Using Function Approximators*, CRC Press, Taylor & Francis, Boca Raton FL, 2010
14. *Manifest programowania zwinnego*, agilemanifesto.org [dostęp 2022-12-03]

## Artificial intelligence in modelling of aerospace objects or agile modelling

Modeling of flying objects is essential in the design and construction of all kinds of simulators. The aim of modeling is to simulate the behavior of objects in such a way that it is acceptable to pilots. Classical modelling, based on the use of equations of classical mechanics (Newton's principles of mechanics: theorems on the change of momenta and others) and the use of basic aircraft data as well as the results of flight tests turns out to be insufficient. Simulators built in this way are usually assessed by pilots as not meeting training requirements. This is especially true for high maneuvering aircraft and helicopter simulators.

The paper presents a new approach to the modeling of flying objects consisting in construction of a hybrid model that combines the classical model with formal models based on neural networks. The AI-derived concept of reinforcement learning was used to build and tune such models. The novelty is that the learning is based on pilots' assessments which are qualitative, often imprecise and very dependent on the pilot and his current psychophysical state. Learning utility functions is therefore defined in a fuzzy form. This required development of dedicated numerical methods that could be used in an interactive

mode because conditions of realization of projects (operations on the facility under construction with limited availability of test pilots and with short deadlines) exclude the possibility of using standard off-line procedures.

The realities of developing hybrid models are in line with the basic theses of agile programming, so the name agile modeling was proposed for such a modeling by analogy. The results of the application of agile modeling in practice for construction of an advanced high-maneuver aircraft simulator are presented. It is shown that the use of agile modeling results in very good flight characteristics of the modelled aircraft, much better than those obtained by identification of the aircraft model on the basis of flight tests.