

REPÜLŐGÉPEK SZIMULÁTOROS VIZSGÁLATA

ÖSSZEFOGLALÁS

A Budapesti Műszaki és Gazdaságtudományi Egyetem Repülőgépek és Hajók Tanszékének repülésszimulátorán (2. Ábra) egy MATLAB alapú, repülésmechanikai modellek és robotpilóta rendszerek tesztelésére alkalmas szoftverkörnyezetet alakítottunk ki. Ebben a környezetben egy különleges kialakítású repülőgép (3. Ábra) vezethetőségét vizsgáltuk két különböző, egyszerű felépítésű szabályzórendszerrel. A virtuális tesztrepülések (5. Ábra) alapján értékeltük a szabályzókat, valamint számszerűen kimutattuk a vezetési rutin szerepét is.

Kulcsszavak: repülésszimulátor, MATLAB, Aerosim, virtuális tesztrepülés, szabályzó

BEVEZETÉS

A Budapesti Műszaki és Gazdaságtudományi Egyetem Repülőgépek és Hajók Tanszékén egy oktatási és tudományos célokra alkalmas repülésszimulátort építettünk. Úgy építettük meg a szimulátort, hogy alkalmas legyen több független szoftver-konfigurációban is működni. Amikor elkészült még csak egyetlen szoftverkörnyezet állt rendelkezésünkre, mely a Microsoft Flight Simulatorra és számos kiegészítőprogramra támaszkodott. Habár ez a konfiguráció nagyon alkalmas oktatási, bemutatási célokra, új repülési eljárások tesztelésére vagy a pilóták viselkedésének vizsgálatára, a program nem teszi lehetővé a repülési modellt, a számítási algoritmusok módosítását. Ezért szükségünk volt egy, ilyen téren, kifinomultabb platformra, ahol repülési modelleket tudunk építeni és különböző robotpilóta rendszereket tudunk tesztelni. A modellépítéshez egy, az iparban is széles körben elterjedt alkalmazást, a MATLAB-ot választottuk. A megoldandó probléma az volt, hogy hogyan tudjuk a szimulátorunk hardver jeleit bejuttatni a MATLAB-ba, illetve, hogy a számítási eredményeket hogyan tudjuk megjeleníteni.

Először a repülés szimulátor kerül bemutatásra, ezt követi majd a MATLAB-os szoftverkörnyezet kialakításának taglalása, végül a rendszer egy alkalmazásának leírása következik.

A REPÜLÉSSZIMULÁTOR LABORATÓRIUM

A repülés szimulátor a Boeing 737 NG típusú repülőgép pilótafülkéjének mintájára készült, a speciális kutatási és oktatási igények miatt kisebb-nagyobb változtatásokat alkalmaztunk. A Boeing 737 egy széles körben elterjedt, közepes hatótávolságú, két hajtóműves utasszállító repülőgép (1. Ábra).

Az összes fontosabb kormány szerv kialakításra került a szimulátorban. A bal oldali, kapitányi ülésből a pilóta egy szarvkormányval irányíthatja a repülőt, a legtöbb repülőgépen ilyen kormány-kialakítás található, beleértve a 737-est is. A jobb oldalon viszont az Airbus gépekből ismert sidestick-et alakítottunk ki, így lehetőségünk nyílik összehasonlítani a két repülőgép-vezetési filozófiát, az Airbus-féle sidestick-ot és a Boeing-féle szarvkormányt (2. Ábra). Négy nagy méretű LCD képernyő található a pilótafülkében, ezekre tudjuk kivetíteni a repüléshez szükséges műszereket. Az egyik képernyő érintő-képernyőként is üzemelhet, ez a kialakítás lehetővé teszi új típusú műszerek fejlesztését és tesztelését.



1. Ábra: Boeing 737 NG típusú repülőgép

A szimulátort hat személyi számítógép vezérli, melyek helyi hálózatban működnek. Az egyszerű PC-k használata, célszámítógépek helyett, tette lehetővé a nagyfokú rugalmasságot a szoftverkörnyezet kialakításában. Jelenleg három független konfiguráció áll a rendelkezésünkre, mindegyik különböző feladatok ellátására lett optimalizálva. Az első kialakítást demo verzióknak neveztük el, ez a Microsoft Flight Simulator 2004-es programra és számos kiegészítő alkalmazásra épül. Ezt a változatot főleg az oktatásban és bemutatók alkalmával használjuk. A második a tudományos verzió, mely a MATLAB szoftver köré épül, a cikk további részében részletesebben esik róla szó. Végül a harmadik kialakítás a fejlesztői verzió, ebben az FLSIM és a VAPS szoftvereket alkalmaztuk, ezek a szoftverek teszik lehetővé új típusú kijelzők, műszerek megalkotását és tesztelését.



2. Ábra: A pilótafülke kialakítása

A SZIMULÁCIÓS PROGRAMKÖRNYEZET

Egy olyan programkörnyezet kialakítását tűztük ki célul, amiben lehetőségünk van különböző repülésmechanikai modellek építésére és ezek a modellek virtuális tesztrepülés formájában kipróbálhatóak a repülés szimulátorban. Erre a feladatra a MATLAB/Simulink alkalmazást választottuk, mely széles körben ismer a tudományos és az ipari életben. Három fő problémával szembesültünk a munka során:

- Hogyan tudjuk a szimulátor hardver elektromos jeleit egy Simulink modellbe bevezetni?
- Hogyan alakítsuk a számítási eredményeket pilóta szemszögű külső képpé?
- Hogyan végezzük a számításokat valós időben?

A lehető legegyszerűbb módot kerestük, hogy megoldjuk ezeket a problémákat. Az összes problémára találtunk megfelelő megoldást, ráadásul csak ingyenes, az internetről letölthető szoftvereket alkalmaztunk.

Az első és a második probléma megoldásához az Unmanned Dynamics cég Aerosim blockset-jét [1] használtuk, mely ingyenes oktatási célokra. Ez a blockset tartalmaz egy joystick interfészt, mely képes a Windows-ban definiált első joystick jeleinek beolvasására. Szimulátorunk egy úgynevezett EPIC kártyát tartalmaz, ennek segítségével a potenciométerek jeleit a számítógép számára elérhetővé tudjuk tenni, úgy, hogy azok úgy jelenjenek meg, mintha egy joystick lenne a számítógépre kötve. Tehát az első problémát sikerült megoldani az interfész felhasználásával és az EPIC kártya megfelelő programozásával.

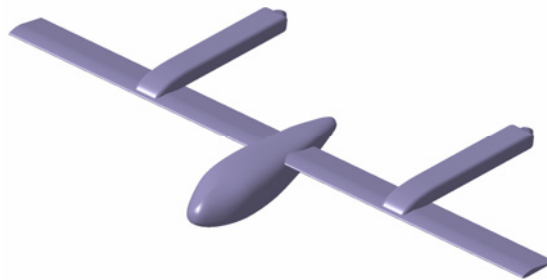
A külső kép generálásához egy ingyenes szimulátor szoftvert használtunk, a Flightgear Flight Simulatort [2]. A fejlesztők felkészítették a programot külső jelek fogadására és megjelenítésére. Az Aerosim Blockset erre a feladatra is tartalmaz egy interfészt, melynek segítségével a Simulink számítási eredményeit a Flightgear-nek tudjuk továbbítani megadott időközönként.

A harmadik probléma különböző módokon oldható meg, mi az IOLib blocksetet [3] hívtuk segítségül. Az IOLib blockset tartalmaz egy speciális blokkot, az RTCsim-et, ha ezt a blokkot a Simulink modellbe helyezzük a számítást megpróbálja majd szinkronban tartani a valós idővel, vagyis a szimulációban eltelt egy másodperc a valóságban is egy másodpercet fog jelenteni. A szimuláció úgynevezett soft real time módban fog futni, mert a szinkronizálást kis hibával lehet csak megvalósítani. Az egyetlen hátránya ennek a megoldásnak, hogy csak akkor működik, ha a számítás a blokk nélkül gyorsabb lenne, mint ahogy valóságban az idő telik.

A tesztrepülésekhez a pilótának szüksége van műszerekre. Jelenleg a Flightgear szoftverbe beépített, könnyen programozható head-up displayt használjuk, ez a legfontosabb repülési paramétereket a külső kép elé vetíti. A kijelzett paraméterek: magasság, sebesség, irányzög és függőleges sebesség.

Ez a szoftverkönyezet két PC-t használ fel a hatból. Az első PC-n a MATLAB alkalmazás fut, ami a modell számítását végzi. A második számítógépen pedig a Flightgear szimulátor program fut, ami a kapott jelek alapján megjeleníti a külső képet és a projektoron keresztül kivetíti a vászonra.

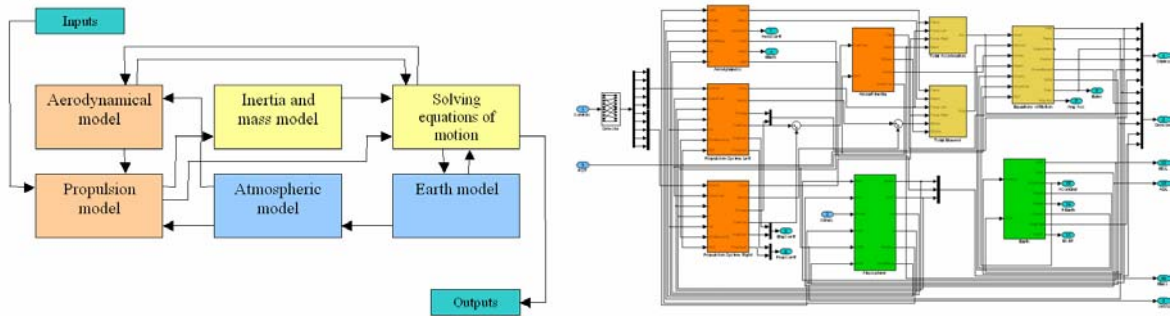
KÜLÖNLEGES KIALAKÍTÁSÚ REPÜLŐGÉP MODELL ÉS VIRTUÁLIS TESZTREPÜLÉSE



3. Ábra: Különleges kialakítású UAV [4]

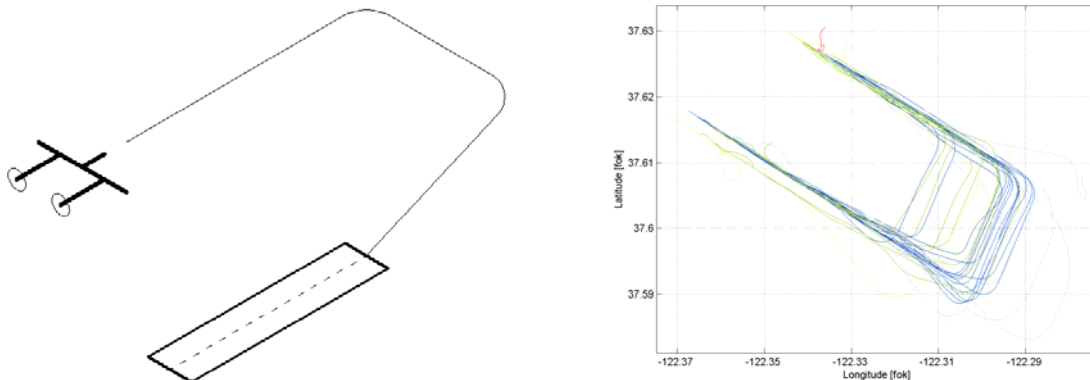
Az előzőekben ismertetett tudományos szoftverkonfigurációt felhasználva egy különleges kialakítású, a valóságban nem létező pilótánélküli repülőgép (UAV) (3. Ábra) [4] viselkedését vizsgáltuk. A tesztelt UAV legfontosabb jellemzője, hogy nem rendelkezik vízszintes és függőleges vezérsíkokkal, emiatt megfelelő szabályzó nélkül szinte lehetetlen volt vezetni a kialakításból származó instabilitás miatt. Ráadásul nem rendelkezik a vezetéshez szükséges, hagyományos, aerodinamikai kormányfelületekkel sem, mint a legtöbb légitármű, hanem tolóerővektor vezérléssel lett ellátva. A tolólégcsavarok forgássíkja dönthető, így hozhatóak létre a kormányzáshoz szükséges nyomatékok. A repülésmechanikai modell MATLAB/Simulinkben lett elkészítve (4. Ábra) az Aerosim Blockset segítségével, a modellen elvégeztük az előző fejezetben részletezett kiegészítéseket. A repülőgép stabilizálására és vezethetővé tételére két különböző szabályzót alakítottunk ki, ezeket a modell linearizálásából nyert állapotter modell felhasználásával terveztük meg. Az első egy nagyon egyszerű, tapasztalati úton megalkotott szabályzó volt, mely az állapotvektorok csak két elemét csatolta vissza, a függőleges és a kereszt tengely körüli szögsebességet (egyszerű

szabályzó). A második szabályzó egy kifinomultabb lineáris kvadrátikus optimalizálási (LQR) elven megtervezett kontroller volt, ez a teljes állapotvektort visszacsatolta a bemenetre (LQ szabályzó).



4. Ábra: A repülésmechanikai modell blokkvázlata és MATLAB/Simulink reprezentációja [4]

A virtuális tesztrepüléshez a repülés szimulátor laboratóriumot egy földi telepítésű vezérlő központként használtuk. A repülés egy egyszerű, iskola-körhöz hasonló repülési útvonalból állt (5. Ábra) [4], tartalmazott egy hosszúfalat, melyen az egyenes-vízszintes repülést lehetett vizsgálni, két jobb fordulót valamint egy kontrollált süllyedést, egy leszállópálya megközelítést. A tesztrepüléseket (5. Ábra) [4] két pilóta végezte, egy tapasztaltabb és egy tapasztalatlanabb. Két repülés történt kikapcsolt szabályzórendszer mellett. Ezek során egyik pilóta sem volt képes megőrizni uralmát a gép fölött. Bebizonyosodott tehát, hogy a repülőgép akkora instabilitással rendelkezik, mely mindenképpen szabályzórendszer jelenlétét teszi szükségessé.



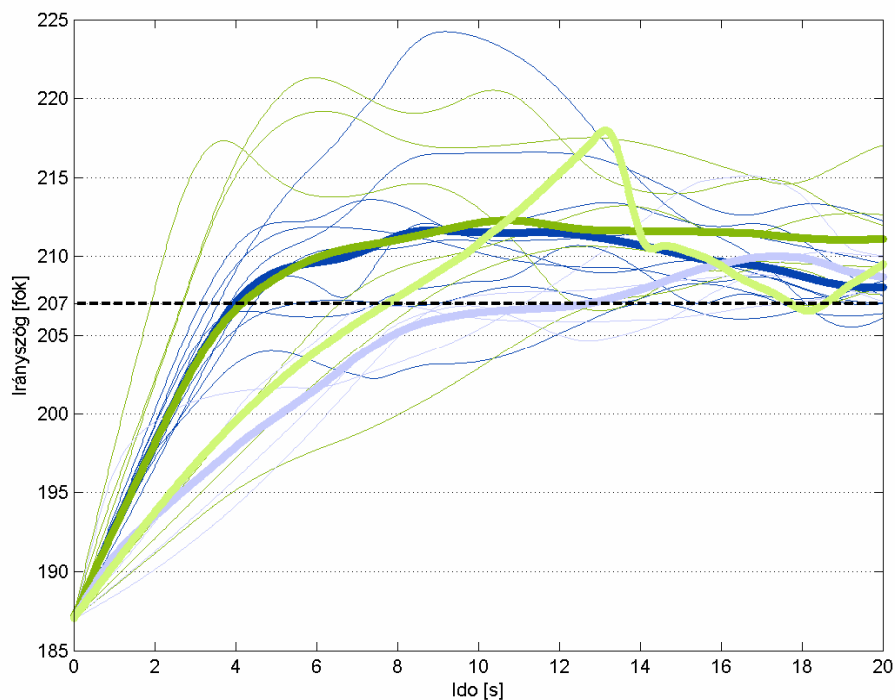
5. Ábra: A tervezett és a tesztek során teljesített útvonal [4]

A kiértékeléshez két statisztikai módszert használtunk, vizsgáltuk, hogy a megadott feladatokat a pilóták végre tudják-e hajtani az adott szabályzóval (sikeresség vizsgálat) (1. táblázat) valamint a sikeresen végrehajtott feladatok végrehajtási pontosságát is vizsgáltuk (referencia értéktől való eltérés) (6. Ábra).

	Egyenes, vízszintes repülés		Forduló (egy repülésen belül egy vagy kettő)		Besiklás és leszállás		Teljes feladatsor	
	Siker/ Összes	%	Siker/ Összes	%	Siker/ Összes	%	Siker/ Összes	%
Kontroller nélkül, tapasztalt pilóta	0/2	0%	-	-	-	-	0/2	0%
Tapasztalatlan pilóta, egyszerű szabályzó	4/5	80%	4/7	57%	0/1	0%	0/5	0%
Tapasztalt pilóta, egyszerű szabályzó	6/6	100%	9/11	82%	3/4	75%	3/6	50%
Tapasztalatlan pilóta, LQ szabályzó	7/7	100%	14/14	100%	4/7	57%	4/7	57%
Tapasztalt pilóta, LQ szabályzó	14/14	100%	28/28	100%	14/14	100%	14/14	100%

A sikeresség vizsgálatból (1. táblázat) [4] kiderült, hogy az egyszerű szabályzóval a tapasztalatlan pilóta egyszer sem tudta végrehajtani a teljes feladatsort, míg a tapasztalt pilóta a repülései 50%-ában sikeres volt. LQ szabályzóval a sikeresség 57%-osra és 100%-osra növekedett. A kiértékelés tehát megmutatta, hogy ugyan az egyszerűbb szabályzó vezethetővé tette a gépet, szükség van a bonyolultabb LQ szabályzóra. Ebből a kiértékelésből, mintegy melléktermékként számszerűen is ki tudtuk mutatni a pilóták vezetési tapasztalatát.

A forduló kiértékelése reprezentálja itt a referencia követési vizsgálatot (6. Ábra). A következő ábrán a vékony vonalak a tényleges repülések adatai, a vastag vonalak pedig a négy konfigurációhoz tartozó, átlagolással képzett görbék. Leolvasható, hogy LQ szabályzóval mindkét pilóta nagyobb szögsebességgel tudta végrehajtani a fordulót. A kiértékelésben fontos szerepet játszott a pilóták szóveges beszámolója a repülőgép viselkedéséről: Egyszerű szabályzónál csak erősen csúszó fordulókat lehetett végrehajtani, melyekhez az oldalkormányt kellett használni. A csűrő megszokott kitérítése csökkentette a fordulás szögsebességét. Egy bizonyos bedöntési szög fölött a repülőgép elvesztette vezethetőségét és lezuhant. LQ szabályzó mellett a csűrőkormányval lehetett végrehajtani a fordulót. Ebben a konfigurációban könnyebbnek bizonyult a paraméterek tartása a forduló közben.



6. Ábra: A fordulók kiértékelése

ÉRTÉKELÉS ÉS KÖVETKEZTETÉS

A MATLAB és a szimulátor hardver összekapcsolásával egy nagyon sokoldalú kutató környezetet sikerült kialakítani, melyen valós vagy még nem létező repülőgépek modelljeit építhetjük meg és repülhetünk velük, vagy éppen adott repülőhöz tervezhetünk szabályzó és robotpilóta rendszereket. A rendszert folyamatosan, hallgatók bevonásával fejlesztjük tovább.

A tárgyalt eljárás legfontosabb előnyei:

- A MATLAB programot használja, ami széles körben elterjedt.
- Ingyenes kiegészítő szoftvereket használ
- A megépített modell moduláris felépítésű
- A modellépítési eljárás öndokumentáló
- Könnyen módosítható, testreszabható
- Az adatok könnyen kinyerhetők és analizálhatóak
- Csak a komputer számítási kapacitása szab határt

FELHASZNÁLT IRODALOM

- [1] Unmanned Dynamics: Aerosim Blockset Version 1.1 User's Guide, www.u-dynamics.com, 2003.
- [2] FlightGear 0.9.2: www.flightgear.org, 2003.
- [3] ZIMMERMANN Werner: IOlib readme: IOlib.htm, www.it.fht-esslingen.de/~zimmerma/software, 2003.
- [4] PATYI Balázs: Instabil repülőgép repülésmechanikai modellje és vezethetőségének vizsgálata, BME RHT, Budapest, 2004.