SZÍNKÓD

Olyan fejezet, ami átfogalmazást igényel, egy körvonalat ír le

Szabály: Megfogalmazása után vagy kék flag

vagy flag nélkülivé kell válnia

Nem kiforrott, átfogalmazást igényel(het), felül kell vizsgálni

Szabály: 2 iteráció után kerülhet csak le róla a flag

Plágiumot tartalmaz (Le kell hivatkozni / át kell írni)

Szabály: Hivatkozás, kitörlés vagy teljes átfogalmazásig

NEM kerülhet le róla a flag

Kérdés, hogy hivatkozzak le egy törvényt.

Fedlap

Feladat

HALLGATÓI NYILATKOZAT

Alulírott hallgató kijelentem, hogy a szakdolgozat / diplomamunka saját munkám eredménye, a felhasznált szakirodalmat és eszközöket azonosíthatóan közöltem. Az elkészült szakdolgozatomban / diplomamunkámban található eredményeket az egyetem és a feladatot kiíró intézmény saját céljára térítés nélkül felhasználhatja.

Budapest, 20….................................

...............................................

hallgató aláírása

**KONZULTÁCIÓS NAPLÓ**

Hallgató neve: Neptun kód: Tagozat:

Albert Dávid H1B5EF nappali

Telefon: Levelezési cím (pl: lakcím):

+36309067746 2120 Dunakeszi Toldi utca 47/a

Szakdolgozat / Diplomamunka[[1]](#footnote-2) címe magyarul:

Épületen belüli helymeghatározás

Szakdolgozat / Diplomamunka[[2]](#footnote-3) címe angolul:

Indoor location tracking system

Intézményi konzulens: Külső konzulens:

Prof. Dr. Kozlovszky Miklós

Kérjük, hogy az adatokat nyomtatott nagybetűkkel írja!

|  |  |  |  |
| --- | --- | --- | --- |
| **Alk.** | **Dátum** | **Tartalom** | **Aláírás** |
| **1.** | **2022. 03. 22** | **Technológiai lehetőségek áttekintése,**  **feladatok egyeztetése, ütemterv készítése** |  |
| **2.** | **2022. 04. 05** | **Témán belüli irány meghatározása** |  |
| **3.** | **2022. 04. 20** | **Dokumentum formázásának javítása** |  |
| **4.** | **2022. 05. 02** | **Megvalósítási tervezet, összegzés táblázat**  **javítása és kiegészítése** |  |

A Konzultációs naplót összesen 4 alkalommal, az egyes konzultációk alkalmával kell láttamoztatni bármelyik konzulenssel.

A hallgató a Szakdolgozat / Szakdolgozat I. / Szakdolgozat II. / Diplomamunka II. / Diplomamunka III. / Diplomamunka IV. / Projektlabor 2. / Projektlabor 3. / Záródolgozati projekt [[3]](#footnote-4) tantárgy követelményét teljesítette, beszámolóra / védésre [[4]](#footnote-5)bocsátható.

**Javasolt érdemjegy:** ……………………

Intézményi konzulens

Budapest, 2022. 05.13.

Kivonat

Abstract

**Tartalomjegyzék**

[Bevezetés 2](#_Toc149406396)

[1. Alapok 4](#_Toc149406397)

[1.1. Arkhimédész elve(Archimedes’ Principle) 4](#_Toc149406398)

[1.2. A felhajtóerő változásai különböző vízviszonyok között 5](#_Toc149406399)

[1.3. A stabilitás és a felhajtóerő középpontjának a szerepe 5](#_Toc149406400)

[1.4. ? Képletek a területhez, a mozgás, a középponthoz, a tehetetlenségi pillanathoz és a sugárhoz ? 6](#_Toc149406401)

[1.5. ? Transverse and longitudinal stability ? 6](#_Toc149406402)

[1.6. Submarine Maneuvering and Control 6](#_Toc149406403)

[1.7. A víz alatti meghajtás és a légellenállás dinamikája 6](#_Toc149406404)

[2. Üzemeltetés és technológia 7](#_Toc149406405)

[2.1. Szenzorok és műszerek 7](#_Toc149406406)

[2.2. Kommunikációs technikák 7](#_Toc149406407)

[2.3. Emberi interakció 7](#_Toc149406408)

[2.4. Something about basic display parts 8](#_Toc149406409)

[2.5. Path planning 8](#_Toc149406410)

[3. Elérhető szimulátorok 9](#_Toc149406411)

[3.1. UWSim 9](#_Toc149406412)

[3.1.1. Működése 9](#_Toc149406413)

[3.1.2. Előnyei 10](#_Toc149406414)

[3.1.3. Hátrányai 10](#_Toc149406415)

[3.2. ? Ardusub ? 11](#_Toc149406416)

[3.3. UUVSimulator 11](#_Toc149406417)

[3.4. Aqua Underwater Simulator 11](#_Toc149406418)

[3.5. Stonefish 11](#_Toc149406419)

[3.6. HoloOcean 11](#_Toc149406420)

[4. Szimulátor létrehozására alkalmas szoftverek 12](#_Toc149406421)

[4.1. Unreal Engine 12](#_Toc149406422)

[4.2. Unity 12](#_Toc149406423)

[4.3. Gazebo 12](#_Toc149406424)

[5. Basic architecture 13](#_Toc149406425)

# Bevezetés

A tengeralattjárók, az autonóm tengeralattjárók (AUV-k) és más mélytengeri járművek az óceánok, tengerek, folyók és más vízi környezetek titokzatos mélységeiben játszanak fontos szerepet. Ezek a kifinomult gépek és eszközeik nélkülözhetetlen szerepet játszanak a felderítő tevékenységek, tengeri kutatások, környezetvédelem és haditengerészet terén, valamint olyan katasztrófák utáni felderítési és mentési műveletek során, ahol emberi beavatkozás kivitelezhetetlen, vagy igen magas kockázatokkal járna.

Ezen járművek célja olyan feladatok ellátása, amelyre más vízi járművek vagy búvárok beavatkozása nem elégséges vagy lehetséges. Példaként említhetjük a tudományos kutatásokat és környezetvédelmi feladatokat, ahol ezek a gépek a tengerfenék térképezésére, olajszivárgások detektálására és tengeri élőlények megfigyelésére használatosak.

A mélységi vízi járművek izgalmas kihívásokat támasztanak a mérnökök és tudósok előtt, mivel a mélytengerben való működtetés rendkívül komplex és veszélyekkel teli feladat. Ilyen egyedi jellemzők például a mélytengeri nyomásoknak való ellenállás, a korlátozott energiatartalék, a kommunikációs nehézségek, és a pontos navigáció a víz alatti akadályok között. Emiatt ezen járművek tervezése, fejlesztése, tesztelése és üzemeltetése komoly megpróbáltatásokkal jár.

Ezen járművek tervezése és működtetése kritikusan fontos, mivel egy tervezési hiba akár a jármű elvesztéséhez is vezethet, illetve a mélytengeri jármű és a hiba típusától függően emberi életek is veszélybe kerülhetnek. Ezért elengedhetetlen, hogy ezeket a járműveket először szimulációkban és teszteken keresztül vizsgáljuk, mielőtt valós működésükre bíznánk őket. A szimulációk lehetővé teszik számunkra, hogy azonosítsuk és javítsuk a tervezési hibákat, és biztosítsuk a járművek biztonságos és hatékony működését a mélytengeri környezetben.

Fontos megjegyezni, hogy a szimulációk és virtuális környezetek kulcsfontosságú szerepet játszanak az AUV-k és UUV-k fejlesztésében és tesztelésében. A különböző szimulációs megoldások lehetnek segítségünkre a tervezési és tesztelési folyamatok hatékonyabbá tételében, és megkönnyíthetik az AUV-k és UUV-k környezetbarát és hatékony működését. Ebben a kontextusban megvizsgáljuk az elérhető, alkalmas szimulátorokat és mérlegeljük, hogy melyeket érdemes felhasználni, továbbfejleszteni, vagy akár egy teljesen új szimulátort létrehozni a megfelelő tesztelés és fejlesztés érdekében. Az elképzelt szimulátor megalkotása során szükséges figyelembe vennünk a motor vagy szoftverrendszer kiválasztását, amely a legjobban illeszkedik a céljainkhoz és lehetővé teszi a valósághű szimuláció megvalósítását.

Az Óbudai Egyetem BioTech Kutató Központjában zajló tengeralattjáró fejlesztésében veszek részt, elsődlegesen, mint a tengeralattjáróhoz való szimulátor elkészítésében, tesztelésében.

Ebben a bevezetésben áttekintettük a tengeralattjárók, AUV-k és más mélytengeri járművek sokoldalú alkalmazását és az ezekkel járó kihívásokat a vízi környezetek mélységeiben. Emellett hangsúlyoztuk a szimulációk és virtuális környezetek szerepét a járművek tervezési és tesztelési folyamataiban. Meggyőződtünk arról, hogy ezek a technológiák létfontosságúak a mélytengeri járművek fejlesztésében és a tengeri mélységek jobb megértésében és védelmében

Az irodalomkutatást követően a dolgozat bemutatja a szimulátorok kialakítását és működésüket.

# Alapok

Az UUV-ök tervezési és fejlesztési lehetőségeik és irányaik megértéséhez tisztában kell lennünk a témához szükséges alapfogalmakkal és a rá vonatkozó fizikai törvényekkel. A fejezet első felében megismerkedhetünk a haditengerészetben gyakran használt kifejezésekkel, tervezéshez elengedhetetlen együtthatókkal és számítási metódusokkal.

A fejezet második felében pedig bemutatásra kerülnek olyan fejlesztési részelemek, amik a kiszolgáló személyzet számára elengedhetetlenek.

has to consider hydrodynamics forces (buoyancy, drag and lift) and fluid inertia

## Arkhimédész elve(Archimedes’ Principle)

Az ókori görög matematikusnak és filozófusnak, Arkhimédésznek tulajdonított Arkhimédész elve a folyadékdinamika egyik alapfogalma, amely megmagyarázza a felhajtóerőt. Ez az elv jelentősen befolyásolja a tengeralattjárók tervezését és működését. Ez a fejezet vizsgálja Arkhimédész elvének a tengeralattjárókra való alkalmazását és vizsgálja mind a felhajtóerő általi lebegési, valamint merülési képességét.

Arkhimédész elve kimondja, hogy „minden folyadékba vagy gázba merülő testre felhajtóerő hat, amelynek nagysága egyenlő a test által kiszorított folyadék vagy gáz súlyával.” (Wikipédia, 2023).

A tengeralattjáróknak két fő módszerük van a mélységük szabályzására: ballasztartályok használata vagy meghajtó mechanizmusok használata.

A hagyományos tengeralattjárók ballasztartályokat alkalmaznak a felhajtóerejük manipulálásához és különböző mélységek eléréséhez. Ezeknek a tartájoknak a vízzel való feltöltésük során a tengeralattjáró tömegének növekedését eredményezi, ami fokozza a lefelé ható gravitációs erőt, és ennek következtében a tengeralattjáró mélyebbre süllyed. BOOK1

Ettől a hagyományos módszertől eltérve azonban, néhány modern tengeralattjáró alternatív meghajtási mechanizmusokat, például elektromos motorokat használ. A motorokkal lefelé vagy felfelé irányuló erők generálásával szabályozva a tengeralattjáró mélységét. Ez a megoldás a ballasztartályok alkalmazásának egy alternatívája, lehetőséget biztosítva egy új, precízebb és gyorsabban reagáló mélységi szint eléréséhez és fenntartásához.

## A felhajtóerő változásai különböző vízviszonyok között

A felhajtóerő megértésében alapvető felismerni, hogy egy tengeralattjáró, egy jelzéssel jelölt folyadékmennyiséget szorít ki a vízvonaláig. Amikor lebeg, ennek a kiszorított folyadéknak a súlya megegyezik a tengeralattjáró és a tartalmának a súlyával. Ha ismerjük a víz tömegsűrűségét p-t, akkor kiszámíthatjuk a kiszorított folyadék súlyát, amelyet eltolódási súlynak, W-nek, hívnak. Ezek alapkán az eltolódási súly a következőképpen számítható ki:

, ahol p a víz tömegsűrűsége, g a gravitációs gyorsulás, és a kiszorított térfogat.

A sűrűség, p. jelentős szerepet játszik a felhajtóerő meghatározásában. Az SI egységekben az édesvíz értéke körülbelül 9.81 ( ) míg a sós víz esetében 10.06 (). A sósvízből az édesvízbe való váltás jelentősen befolyásolhatja a tengeralattjáró merülését a két víz különböző sűrűsége miatt. Ez a jelenség abból adódik, hogy az édesvíz sűrűsége körülbelül 2%-kal kevesebb, mint az óceáni sósvízé. ("Displacement and Center of Buoyancy")

Továbbá, egy lényeges tényező a hőmérsékelt hatása a víz sűrűségére. A számolás során meghatározott alaphőmérséklettől való eltérés befolyásolhatja a felhajtóerőt, ami korrekciót tesz szükségessé a számításoknál.

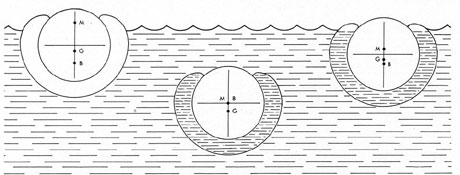
Összefoglalva, a víz sűrűsége, amelyet a sótartalom és a hőmérséklet befolyásol egy tengeralattjáró elmozdulásának alapvető eleme, ami alapját képezi a felhajtóerő számításoknak. Ez elengedhetetlen egy tengeralattjáró tervezéséhez és teljesítményéhez.

## A stabilitás és a felhajtóerő középpontjának a szerepe

A felhajtóerő középpontja kulcsfontosságú fogalom a tengeralattjárók tervezésében. Ez a kiszorított víz tömegközéppontját jelenti, amikor egy tengeralattjáró elmerül vagy lebeg. A félreértések elkerülése érdekében fontos megjegyezni, hogy a felhajtóerő középpontját nem szabad összetéveszteni a víz alá süllyesztett vagy úszó test tömegközéppontjával. (<https://maritime.org/doc/fleetsub/chap5.php>) A felhajtóerő középpontjának jele: B.

A metacentrum egy másik kulcsfontosságú pont a tengeralattjárók stabilitásának elemzésében. Ez az a pont, ahol a test felhajtóerejének középpontján átmenő függőleges egyenes metszi az új felhajtóerő középpontján átmenő függőleges egyenest. A metacentrum pont jele: M és fontos szerepet játszik a test stabilitásának értékelésében. A test gravitációs középpontjának jelölése G. Ha M G felett helyezkedik el, akkor pozitív metacentrikus magasságról beszélünk és a test stabil. Ezzel szemben, ha M G alatt van, a metacentrikus magasság negatív lesz, ami instabillá teszi a testet. (<https://maritime.org/doc/fleetsub/chap5.php>)

A tengeralattjáró stabilitásának mérlegelésekor fontos megjegyezni, hogy a B (felhajtóerő középpontja), G (a gravitációs középpont) és az M (metacentrum) helyzete attól függően változik, hogy a tengeralattjáró a felszínen van-e vagy elmerült. Amikor egy tengeralattjáró teljesen elmerül, B és M egy közös pontban konvergál, míg B fölé emelkedik és elhalad G felett. Amint a ballasztartályokat teljesen elárasztják, B a nyomás alatti hajótest normál felhajtóerejének középpontjába emelkedik, és a stabilitás helyreáll G-vel B alatt. (<https://maritime.org/doc/fleetsub/chap5.php>)

E kulcselemek elhelyezésének gondos ellenőrzése lehetővé teszi a tengeralattjárók számára, hogy elérjék és fenntartsák a kívánt stabilitási szintet.

## ? Képletek a területhez, a mozgás, a középponthoz, a tehetetlenségi pillanathoz és a sugárhoz ?

For a variety of reasons, it is necessary to be able to calculate areas, centroids, volumes-and other geometrical characteristics-of a ship’s form when floating at any prescribed waterline.

## ? Transverse and longitudinal stability ?

## Submarine Maneuvering and Control

## A víz alatti meghajtás és a légellenállás dinamikája

#### A víz alatti mozgásban fellépő ellenállás megértése

#### A víz alatti meghajtás alapelvei

# Üzemeltetés és technológia

## Szenzorok és műszerek

#### Bevezetés a szenzorokba

#### Gyakran használt szenzorok

#### A műszerek kalibrálása és karbantartása

## Kommunikációs technikák

#### Kihívások a víz alatti kommunikációban

#### Akusztikus kommunikációs és modulációs technikák

#### Új technológiák az UUV-kommunikációban

## Emberi interakció

#### Bevezetés az ember-gép interakcióba az UUV-ökné

#### Vezérlő interfészek: Fizikai és szoftveralapú megoldások

#### A hatékony UUV-interfészek tervezési elvei

Delve into the interface and controls allowing operators to interact with UUVs and how this is represented in simulators.

## Something about basic display parts

A sophisticated robot needs a user interface that's intuitive without sacrificing functionality. That's why we are proud of the GUI designed for the #ISE Explorer #AUV.

(<https://twitter.com/IntSubEng/status/1239623766277316608/photo/1>)

## Path planning

Marine Copernicus database

Real time data

Forcasted data

Historical data

# Elérhető szimulátorok

Az első fejezet középpontjában az áll, hogy az UUV-k fejlesztése viszonylag magas költségekkel jár, és a véletlen sérülés vagy elsüllyedés az UUV-k kísérletei vagy tengeri próbái során jelentős veszteségeket okozhatnak. Ebből kiindulva hangsúlyozzuk a hatékony és felhasználóbarát szimulációs rendszerek szükségességét, amelyek segítik az algoritmusok és funkciók ellenőrzését még a tényleges kísérletek vagy tengeri próbák előtt [10,11].

A fejezetben kifejtésre kerül az UUV-k szimulációjának főbb területeit. Emellett bemutatásra kerülnek a már rendelkezésre álló, viszonylag kiforrott szimulációs rendszereket, mint például a UWSim, UUVSimulator, Aqua Underwater Simulator, Stonefish és HoloOcean. Ezekre a szimulátorokra összpontosítva részletes elemzést végzek, kihangsúlyozva előnyeiket és korlátaikat az UUV-k tervezése és tesztelése során.

## UWSim

UWSim, röviden az UnderWater SIMulator, egy olyan szimulációs rendszer, amely a tengeri robotika kutatás és fejlesztés területén használható. Ennek a szoftvernek a segítségével vizualizálható egy víz alatti virtuális scenárió, amelyet standard modellező szoftverek segítségével konfigurálhatunk. Az UWSimbe hozzáadhatunk irányítható víz alatti járműveket, felszíni hajókat, robotikai manipulátorokat és szimulált érzékelőket, amelyeket külsőleg elérhetünk ROS (Robot Operating System) interfészek segítségével. Az UWSim sikeresen alkalmazható víz alatti beavatkozási feladatok logikájának szimulálására és valós beavatkozási feladatok reprodukálására a rögzített naplók alapján.

Az UWSim fejlesztése eredetileg azért indult, hogy egy olyan eszközt nyújtson, amellyel lehetőség van az érzékelési és irányítási algoritmusok tesztelésére és integrálására, mielőtt azokat valós robotokon futtatnánk. Az UWSim fejlesztését az RAUVI és TRIDENT kutatási projektekhez kapcsolódóan indították el.

Az UWSim csak az Ubuntu Linux 9.10 és 12.04 közötti verziókkal kompatibilis, ezért érdemes a használónak az automatikus rendszerfrissítéseket kikapcsolnia.

### Működése

Az UWSim szoftver a vízalatti fizikai szimulációkhoz az Open Dynamics Engine (ODE) nevű fizikai szimulációs motort alkalmazza. Az ODE egy nyílt forráskódú fizikai szimulációs motor, amely C++ nyelven íródott, és rugalmas 3D motorral rendelkezik, amely a jelenet-orientált fejlesztési folyamatokat támogatja. Ennek eredményeként lehetővé teszi a fejlesztők számára, hogy hatékonyan dolgozzanak 3D hardverrel, így könnyebbé és intuitívabbá téve a vízalatti játékok, szimulátorok és bemutatók készítését.

Az UWSim a jelenetek konfigurálásához XML-formátumú dokumentumokat használ, amelyeket egy DTD dokumentum validál. Az Xacro makrók alkalmazását ajánlják a program fejlesztői készülékek, objektumok és járművek könyvtárainak előzetes létrehozásához, hogy elkerülhessük az XML fájlok szerkesztésének körülményességét és hatékonyság hiányát. Ezáltal minimalizálja az XML fájlok időigényes és eredménytelen szerkesztését, miközben elősegíti az elemek újrafelhasználhatóságát és a fejlesztés hatékonyságát.

Az UWSim szoftver az alábbi technológiai megoldásokat alkalmazza:

* Vízalatti jelenetek konfigurálására XML-formátumú dokumentumok segítségével, amelyeket egy DTD dokumentum validál.
* Xacro makrók használatára a hatékonyabb XML-fájlok létrehozásához és könyvtárak kialakításához, minimalizálva az ismétlődő szerkesztést és elősegítve az újrafelhasználhatóságot.
* Óceán jellemzőinek beállítására az "oceanState" blokk segítségével, meghatározva a víz fizikai tulajdonságait a szimuláció során.
* Szimulátor beállítások módosítására a "simParams" blokkban, például shader-ek, felbontás vagy fizikai paraméterek tekintetében.
* Fő kamera paramétereinek beállítására a "camera" blokk segítségével, amely az UWSim főablakában megjelenő jelenetet figyeli és a beállított kamera pozíció és látószöge alapján képes megjeleníteni a kameraképet.
* Vízalatti robotok és szenzorok létrehozására és konfigurálására a "vehicle" címkékkel.
* 3D modellek beillesztésére a jelenetbe és azok interakciójára a "object" blokk használatával.
* ROS interfészek csatolására bizonyos objektumokhoz, robotokhoz vagy szenzorokhoz, egyszerűsítve ezzel a kommunikációt külső vezérlési rendszerekkel.

### Előnyei

* Nyílt forráskódú projekt
* ROS-al összekapcsolható
* Meglévő szimulált járművekkel és környezetekkel rendelkezik

### Hátrányai

* Régi technológiákon alapul
* Környezet telepítési és futtatási nehézségek
* Hiányos, nagyon alapszintű dokumentáció
* Vizuálisan visszamaradott
* Közösségi létszáma alacsony
* Megszűnt a támogatása, fejlesztése (2013)

SAJÁT KÉP A SZIMULÁTORRÓL

## ? Ardusub ?

Megnézni, hogy ez csak egy open source vezérlő program vagy szimulátor funkcióval is rendelkezik esetleg?

Ihletet meríteni az ebben implementált funkciókból.

## UUVSimulator

<https://uuvsimulator.github.io/>

## Aqua Underwater Simulator

<https://github.com/rmartin5/aqua-sim-ng>

<https://ieeexplore.ieee.org/document/5422081>

## Stonefish

## HoloOcean

# Szimulátor létrehozására alkalmas szoftverek

Ebben a fejezetben néhány olyan szoftvert tekintünk át, amelyek alkalmasak szimulátor alkalmazások létrehozására, és amelyek rendelkeznek fizikai motorral, valamint lehetőséget kínálnak az alkalmazások fejlesztésére és testreszabására. A két legkiemelkedőbb platform, amelyeket megvizsgálunk, az Unreal Engine (5) és a Unity. Ezek a szoftverek széles körű eszközöket és funkciókat biztosítanak szimulátorok létrehozásához és testreszabásához, és lehetőséget kínálnak a valósághű fizika modellezésére és a gyors prototípus elkészítéséhez.

Emellett tekintetbe veszünk további olyan szimulációs platformokat is, amelyek szintén alkalmasak saját szimulátorok készítésére, fejlesztésére, például a Gazebo vagy a CARLA. Ezek a szoftverek is lehetőséget biztosítanak az alkalmazások testreszabására és fejlesztésére és specifikus igények kielégítésére is használhatók.

A fejezet során áttekintjük ezeket a szoftvereket, bemutatásra kerülnek az előnyeik és korlátaik, és segítséget nyújtunk a megfelelő platform kiválasztásában annak érdekében, hogy hatékonyan fejleszthessünk saját szimulátort a céljaink eléréséhez.

## Unreal Engine

## Unity

## Gazebo

# Basic architecture

IMAGE OF THE COMPONENTS AND HOW THEY ARE CONNECTED WITH EACH OTHER

1. Megfelelő aláhúzandó! [↑](#footnote-ref-2)
2. Megfelelő aláhúzandó! [↑](#footnote-ref-3)
3. Megfelelő aláhúzandó! [↑](#footnote-ref-4)
4. Megfelelő aláhúzandó! [↑](#footnote-ref-5)