

Navigáció, térképkészítés a kiszolgáló robot számára

RoboCup@Home Education 2022 Webinar alapján

Készítette:

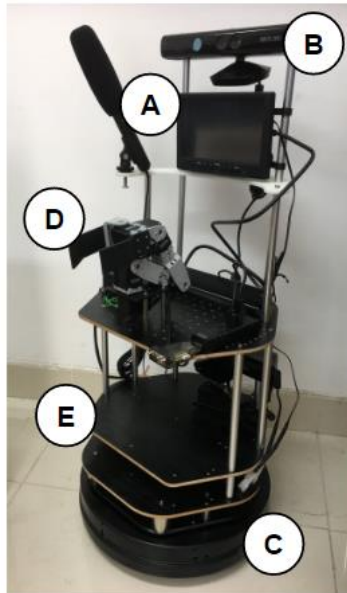
Necze Máté

Robotintelligencia Tudományos Diákkör tagja

Nyíregyházi Egyetem

A kiszolgáló robot fejlesztésében a részfeladatok

Basic Service Robot Components



A. Robot **MOUTH & EARS**

- Human-Robot Interaction

B. Robot **EYES**

- Visual Perception

C. Robot **LEGS**

- Autonomous Navigation

D. Robot **ARM**

- Object Manipulation

E. Robot **BRAIN**

- AI, Machine Learning, Cloud Computing, Big Data

Hardver és szoftver igény a navigációhoz

Hardver:

- Szükségünk van egy laptopra a szimulációhoz.

Szoftver:

- Operációs rendszernek az Ubuntu-t használjuk.
- Robotfejlesztéshez a ROS-t használjuk.
- A szimuláció felépítéséhez a Gazebo szoftvert használjuk.



ubuntu

ROS



Adatgyűjtés eszköze a navigációhoz

A robotunk rendelkezik egy Lidar Szenzorral ami segítséget nyújt a SLAM térképkészítéshez és autonóm navigációhoz.



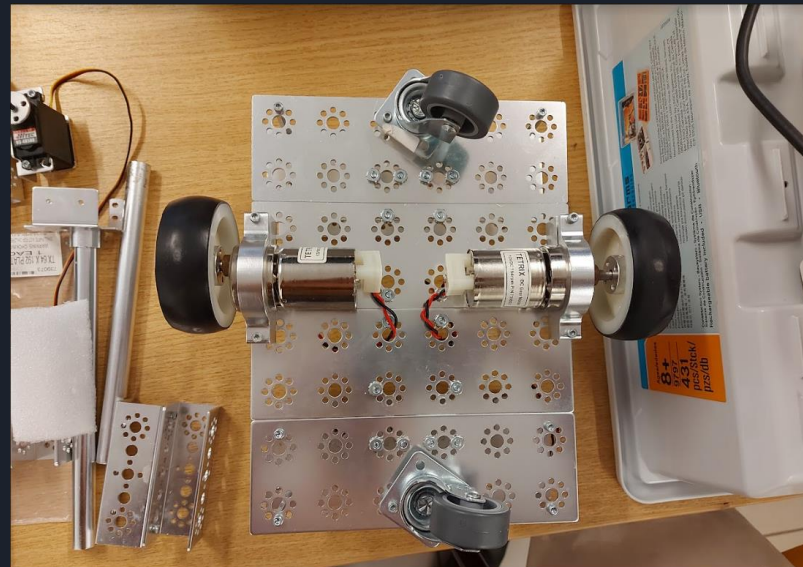
Lézeres adat felhasználás eredménye a baloldali térkép



A Lidar lézerfényt használ a távolságméréshez. A lézer által kibocsátott folytonos elektromágneses hullám kölcsönhatásba kerül a terjedés irányában elhelyezkedő objektumokkal, és visszaverődik azokról.

Kiszolgáló robot meghajtó rendszerének jelenlegi állapota

A robotunk alváza Tetrax készletből épül: 2 db DC motor, hajtja meg, szimmetrikusan két bolygó kerék biztosítja a könnyebb mozgását.



A Turtlebot2 robot mint követendő példa

A célunk olyan térképkészítési és navigációs módszert elérni, ami a Turtlebot 2 példájára épül.

A Webinar oktatója(Jeffrey Tan) javaslata szerint. A RoboCup@Home Education versenyeken legtöbb csapat ezt a alaprobotot használja.

<https://www.turtlebot.com/turtlebot2/>

Legújabb fejlesztései a cégnek:

Turtlebot 4

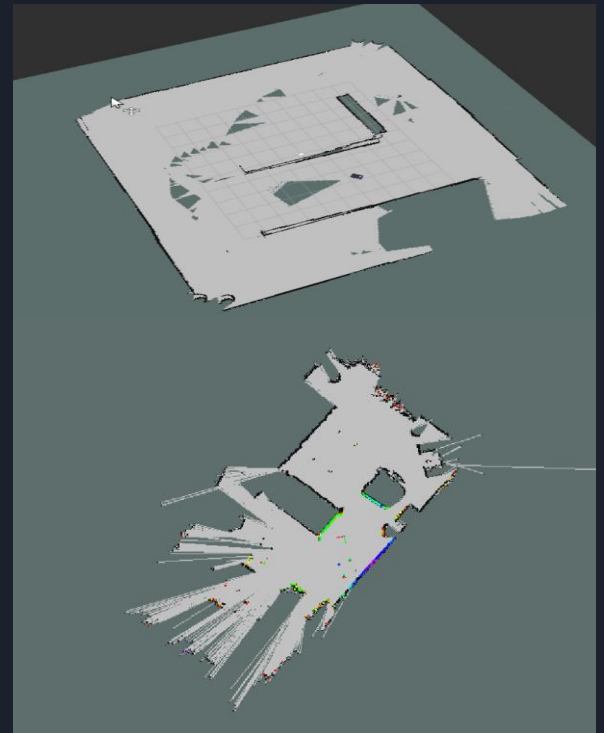
Jelenleg megvásárolható: 2400 Euro-ba kerül



Simultaneous Localization And Mapping(SLAM)

SLAM használathoz telepítenünk kell a “gmapping” ROS csomagot, amely <http://wiki.ros.org/gmapping> weboldalon található és letölthető.

Ez a csomag tartalmaz olyan parancsokat amely segítségével fel tudunk térképezni egy adott teret és ezt le tudjuk menteni egy .yaml típusú fájlba.



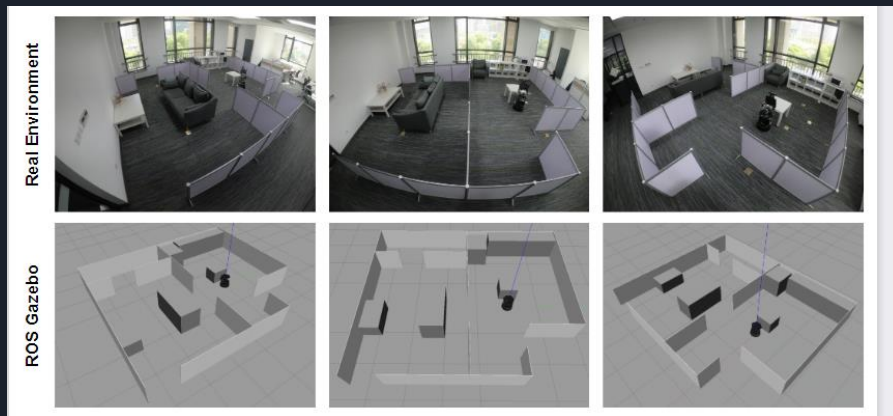
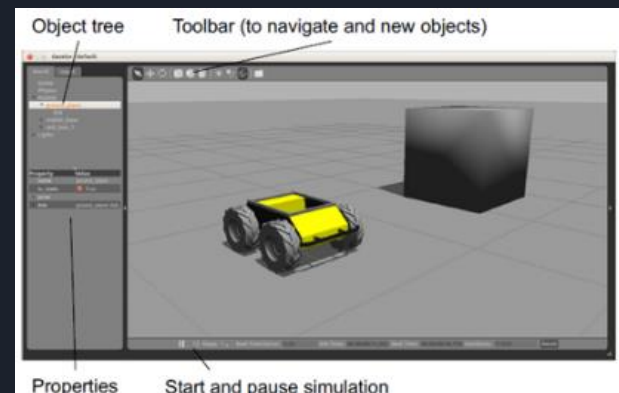
Gazebo Simulator Program

A Gazebo egy nyílt forráskódú 3D robotikai szimulátor program. Amely helyettesítheti a fizikai robotot a tanulás során.

ROS Melodic verziója tartalmazza a Gazebo programot.

Képes olyan robotokat és szenzorokat modellezni, amelyek "látják" a szimulált környezetet, például lézeres távolságmérőket, kamerákat (beleértve a nagy látószögűeket is), Kinect stílusú érzékelőket stb.

A szimuláció fejlesztéséhez a Gazebo-t fogjuk használni.

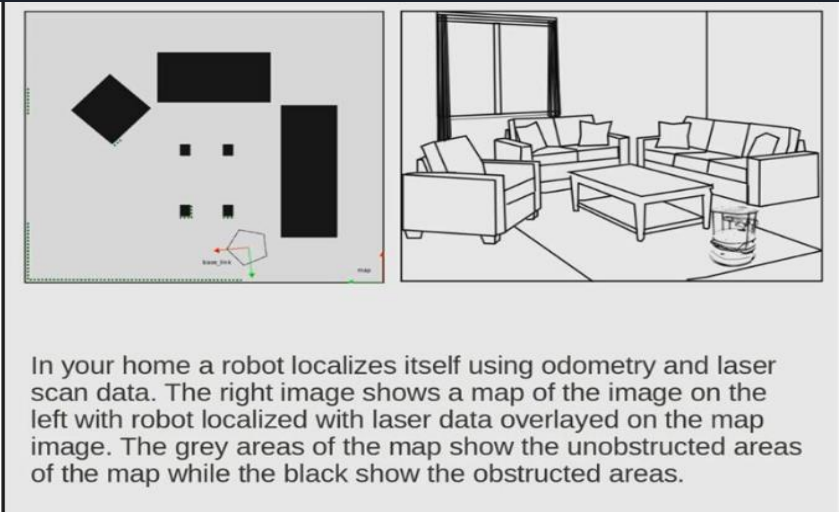


Gazebo használata

Ezen a képen láthatjuk, hogy a robot feltérképezi a területet de a asztaltalnak csak a négy sarkát sikerült, tehát a robot úgy gondolja, hogy ott is tud közlekedni.

Ennek a probléma megoldásához tudjuk használni a Gazebo szoftvert.

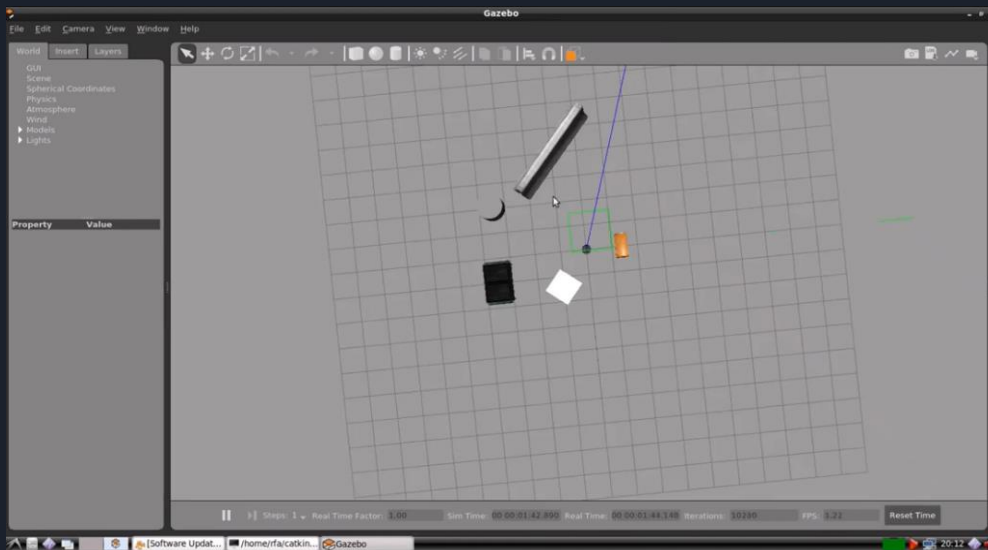
Scannelés után még a térképen tudunk változtatni és megadhatjuk a robotnak , hogy az asztalt teljes egészébe vegye figyelembe.



Szimulációhoz hasznos terminál parancsok

Először létre kell hozni a robot modelljét:

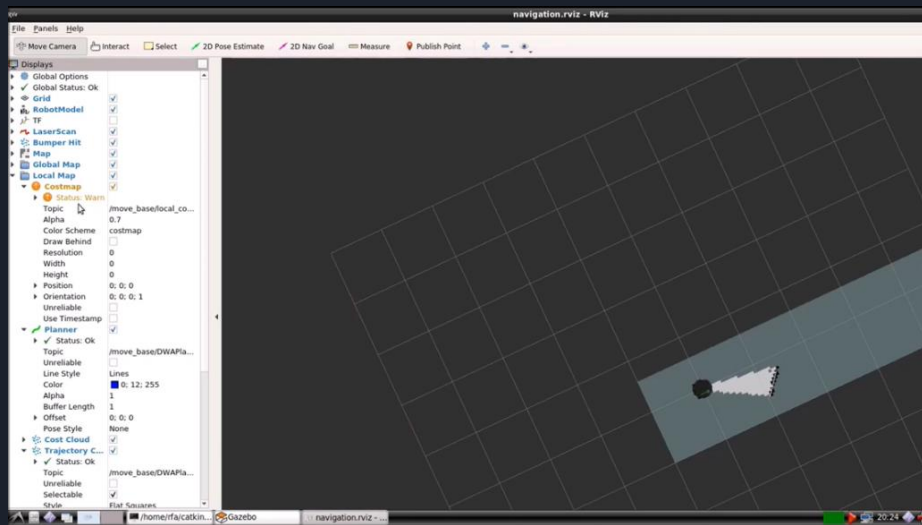
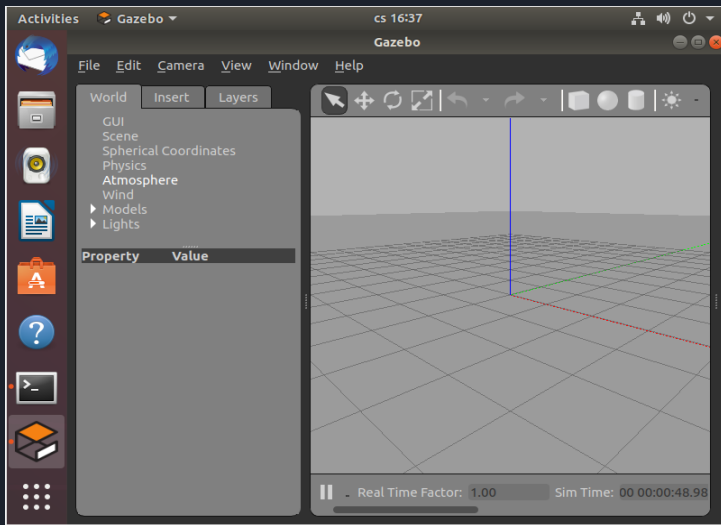
- `roslaunch turtlebot_gazebo gmapping_demo.launch`



Szimulációhoz hasznos terminál parancsok

Rviz elindítása “a vizualizáció”

- `roslaunch turtlebot_rviz_launchers view_navigation.launch`



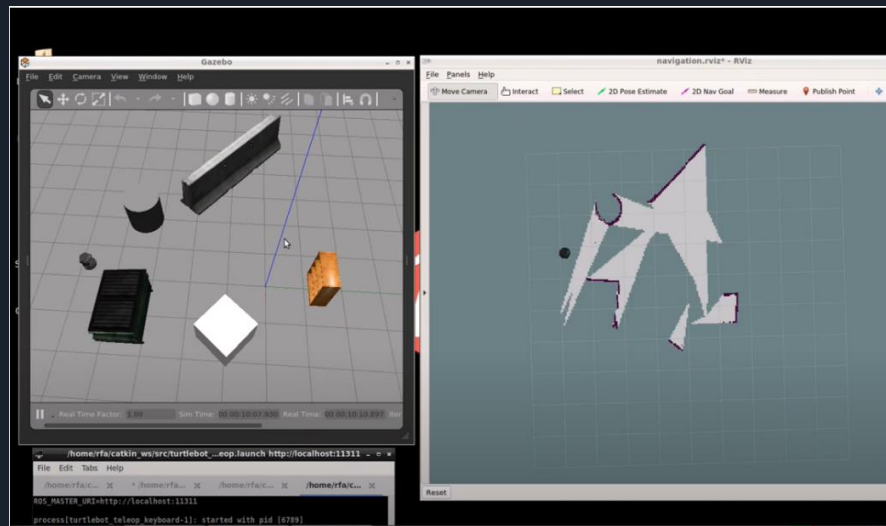
Szimulációhoz hasznos terminál parancsok

Teleop Twist keyborad beállítása

- `roslaunch turtlebot_teleop keyboard_teleop.launch`

Beolvasott térkép elmentés

- `roslaunch map_server map_saver -f /home/<username>/my_map`



Ezen a képen láthatjuk, hogy a robot a objektumok elhelyezkedését és az információkat megjegyzi.

Autonom navigáció

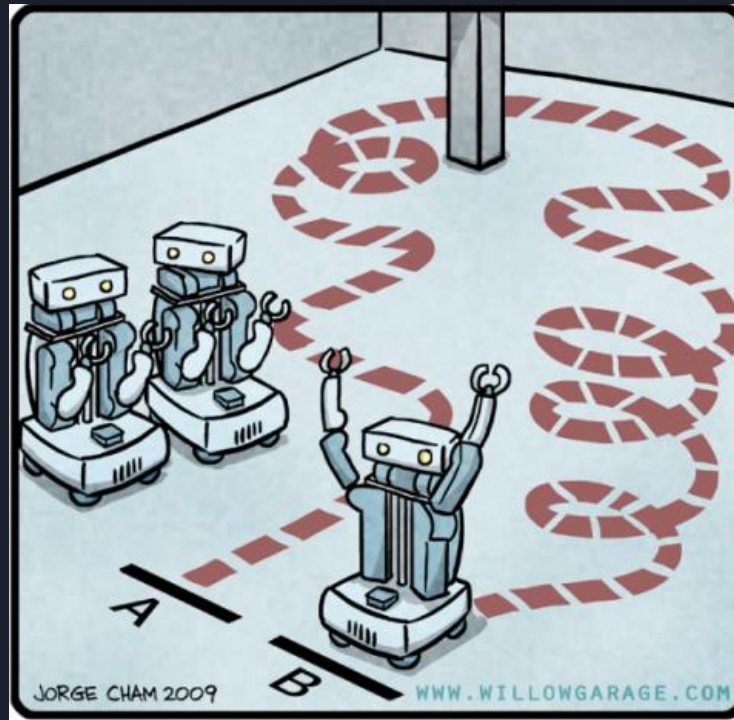
Tekintsünk egy robotot a környezetének meglévő térképével. Amikor a robot mozog, tudnia kell, hol van ezen a térképen.

Helyének és irányszögének meghatározását szenzoros megfigyelések alapján nevezzük robotlokalizációnak.

A lokalizációhoz a “amcl” (Adaptive Monte Carlo Localization) lokalizációs algoritmust lehet használni.

Elérhető:

- <http://wiki.ros.org/amcl>



Autonom navigáció Rviz szimulációival

Képeken láthatjuk a Gazebo-n szimulált robot mozgását(bal oldal).

A lokalizációt a Rviz program képernyőjén látjuk(jobb oldal).

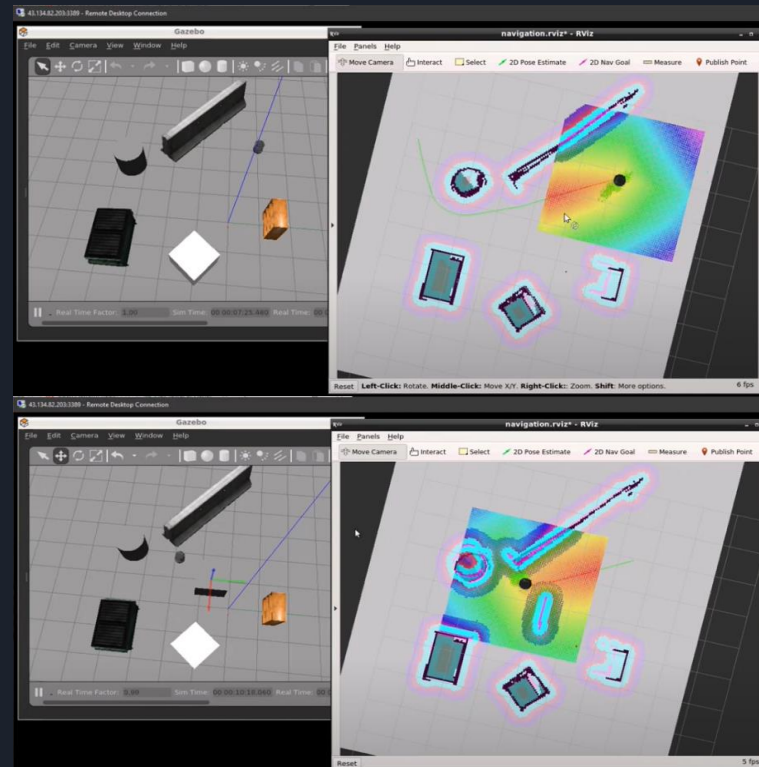
Van egy térképünk és a robotot A pontból B pontba akarjuk juttatni. A robot lokalizációjára van szükség ehhez az amcl rendszert használjuk.

2D Nav Goal nyíl segítségével megadjuk, hogy mi is a célpont ahová szeretnénk a robotot elküldeni.

Itt láthatjuk, hogy az amcl segítségével a robot kikerüli a objektumokat a cél elérése érdekében.

Most hozzáadunk egy új objektumot és láthatjuk, hogy ez a robot észreveszi és ki is kerüli.

Az amcl updatolta a térképet és kikerülte a robot az új akadályt. Ezt nevezzük dinamikus akadály elkerülésnek.





Összefoglalás:

RoboCup@Home Education oktatási projektét nyílt platformu kiszolgáló robot fejlesztéséhez hasznosnak tartottam. A autonóm navigációs rendszer fejlesztésében nagy segítséget nyújtanak a azok ROS program csomagok amelyeknek használatát el lehet sajátítani a Online oktató tanfolyamon való részvétel.

2 fő programra fókuszáltam:

- a térképkészítéshez ajánlott “gmapping” programcsomagra amely a SLAM algoritmust használja.
- Rviz programcsomagra amely a részecskeszűrős “amcl” valószínűségi lokalizációs módszert használja.



Köszönöm a
figyelmet!

Dátum: 2022.12.15