[[1]](#footnote-1)

Kormánymű geometria optimalizáció megerősítéses tanulással

Dóka Tamás, *Phd hallgató, BME*

*Abstract*—

# Bevezetés

A

Mérnöki feladatok nagy részében valamilyen paraméterhalmaztól függő problémát kell optimalizálnunk, nincs ez máshogy a gépészeti tervezésben sem. A tervezés során általában nem állnak rendelkezésre a pontos számításokhoz szükséges adatok, így a tervezés korai fázisában ezeket becsléssel határozzuk meg, majd iteratívan közelítünk az optimális megoldáshoz. Megerősítéses tanulással is egy problémát optimalizálunk, legyen szó a go játék megnyeréséről vagy egy komplex tervezés során felmerülő döntések meghozataláról. A házi feladat során egy négykerekű jármű kormányművének egyszerűsített, kétdimenziós (felülnézeti) reprezentációjának geometriai optimalizációját terveztem megvalósítani megerősítéses tanulás segítségével. A beszámolóban először röviden bemutatom a megerősítéses tanuláshoz használt OpenAI Gym környezet felépítését, az optimalizációs problémát, a tanítás során használt környezetet, az általam implementált tanító algoritmust, végül az eredmények összefoglalását.

# OpenAI Gym megerősítéses tanuláshoz

A megerősítéses tanulás alapja egy környezetből (environment) és egy a környezettel interakciót létesítő ügynökből (agent) áll (1. ábra).



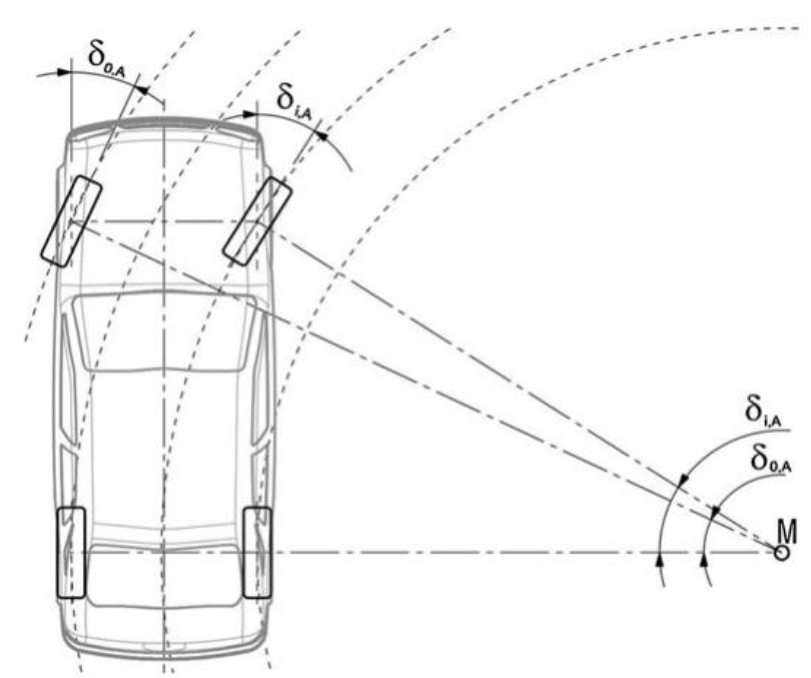
1. Ábra. A környezet és az ügynök kapcsolata

A környezet „tartalmazza” a problémát, amelyet az ügynök igyekszik megoldani a környezeten végzett hatásokkal (actions), amelyek megváltoztatják a környezet belső állapotát (state). Az ügynök által elvégzett hatásra bekövetkezett új állapot mellett az ügynök visszakap egy jutalom (reward) értéket is, ami a hatás „jóságát” jellemzi a környezet kezdeti állapota mellett. Az ügynök célja, hogy minden állapot mellett olyan hatást gyakoroljon a környezetre, ami után a legnagyobb jutalmat kapja meg.

Az ügynök – környezet interakció megvalósítására megalkottak egy „szabványt”, ami lehetővé teszi, hogy egy sztenderd interfészen keresztül kommunikáljon egymással a környezet és az ügynök implementáció, ezt nevezik OpenAI Gymnek. OpenAI Gym környezetekből több szabadon hozzáférhető, így saját ügynök implementációkat lehet rajtuk tesztelni, illetve saját készítésű környezeteket is lehet általános ügynök algoritmusokkal (például stable baselines) tesztelni, ha rendelkeznek gym környezet interfészekkel. Ezek az interfészek többek között a reset(), step(), és render() függvények.

# Az optimalizációs probléma

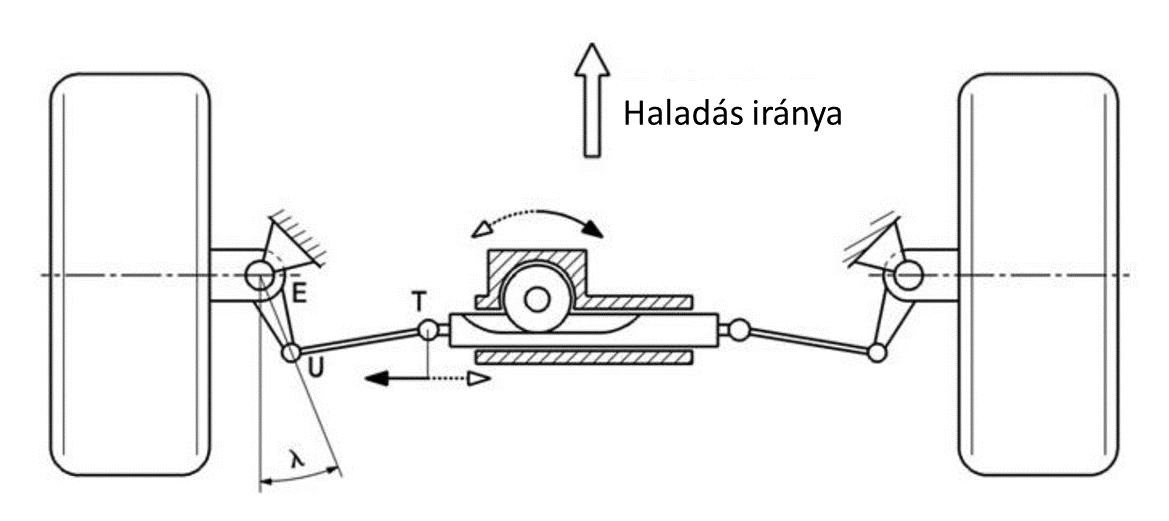
Az optimalizációs probléma egy négykerekű jármű ideális kanyarodásának megvalósítása fix elrendezésű kormánymű geometriával. Ideális, csúszásmentes kanyarodás akkor jön létre, ha a jármű kerekei egy közös középpontból húzott köríveken haladnak kanyarodás során (2. ábra – M pont). Ha csak elsőkerék-kormányzású a jármű, úgy a két hátsó kerék közös tengelye megszab egy egyenest, amin ez a közös középpont elhelyezkedhet. Ha meghatározunk egy fordulókör sugarat, a külső köríven haladó kerék talppontjából ezzel a távolsággal húzott ív, és a hátsó tengely egyenese kiadja a közös középpontot. Ahhoz, hogy az ideális kanyarodás megvalósuljon, a belső keréknek is e középpontból húzott köríven kell haladnia, így minden fordulókör sugárhoz, adott lesz a külső , és a belső kerék , egyeneshez képesti elfordulási szöge (2. ábra).



2. ábra. Egy négykerekű jármű kerekeinek és a fordulókör középpontjának viszonya.

## Kormánymű geometria

A közúti járművek legelterjettebb kormánymű kialakítása az úgynevezett fogaskerék – fogasléc (rack-pinion) elrendezés (3. ábra)



3. ábra. Fogaskerék – fogasléc kormánymű kialakítás

Ebben az egyszerű kialakításban a fogasléc oldalirányban való lineáris mozgatása a külső és belső kerék eltérő mértékű elfordulását eredményezi, így megfelelő méretek megválasztásával létrejöhet az ideális kanyarodáshoz közeli állapot minden fordulókör nagyság mellett. Az optimalizálás során tehát egy adott tengelytáv (wheelbase), nyomtáv (track width), valamint a kerék talppontja és az elfordulási tengelye (3. ábra – E pont) közötti távolság (kingpin distance) mellett kell meghatározni, hogy a forgatókar (control arm) (3. ábra E-U) és kormányösszekötő-rúd (tie-rod) (3. ábra E-T) végpontjai hogyan helyezkedjenek el. További kritérium még egy mindennapi használathoz feltétlenül szükséges minimális fordulókör előírása is.

## Változtatható paraméterek

Az optimalizálás során tehát a változtatható paraméterek a 3. ábrán jelölt T és U pontok x és y irányú koordinátái lesznek. Ha a derékszögű koordináta-rendszer középpontját az első tengely közepére helyezzük és y irány felel meg a menetiránynak.

Ebben az esetben, amennyiben T pont y koordinátája az első tengely mögött helyezkedik el, negatív lesz, a fogasléc pozitív x irányba való mozgatásakor a kerekek a pozitív iránynak megfelelő forgást végeznek (a jármű balra fordul).

A paraméterek beállítása után kiszámítható a fogasléc mozgatásából következő kerék elfordulások mértéke, így a külső köríven haladó kerék elfordulásának függvényében felírható lesz a belső kerék elfordulása. Ez a függvényszerűség az ideális kanyarodásra is felírható, így a kormánymű hibája az ideális görbe és az adott paraméterek mellett létrejött görbe közötti különbség négyzetösszegeként definiálható. Az optimalizálási feladat megoldott, ha ez a hibaérték eléri a nullát, vagy egy általunk előírt határ alá csökken, vagyis a tényleges görbe megközelíti az ideálisat.

# A tanítás során használt környezet

A megerősítéses tanítás megkezdéséhez először az interakcióban résztvevő környezet felépítését kezdtem meg. A környezetnek az ügynök által kiváltott hatásra vissza kell adnia a hatásra kialakult belső állapotát, valamint egy jutalom értéket. A környezetnek továbbá követnie kell a sztenderd gym felépítést is, amelynek a kötelező elemei a következők:

1. . [↑](#footnote-ref-1)