# 22. Tétel

Mobil robotok navigációja akadályok között. Akadályelkerülési módszerek

## Navigáció különböző autonómiájú robotoknál:

### Navigációs szintek

1. **Automatikus mozgásszabályozás**
   1. Csak a helyi szabályozási feladatok ellátása
   2. "távirányítás"
   3. DAS rendszerek (tempomat, menetstabilizátor, ABS, sávtartás)
2. **Autonóm mozgástervezés**
   1. Kognitív döntések: tervezi a saját mozgását az akadályok és a célpont imseretében
   2. Csak a mozgás tervezése tartozik ide, a beavatkozás az ember feladata
   3. GPS nav program -- mozgástervezés
3. **Autonóm térképalkotás**
   1. Alkalmazkodás a környezet változásaihoz
   2. A vezető a navigációs célpontot jelöli ki csak
4. **Teljes autonómia**
   1. Teljes önállóság
   2. A feladat magasabb szinten kerül meghatározásra
   3. Mozgástervezés, végrehajtás, navigációs célpontok kiválasztása a robot feladat

Állapot trajketória: Időfüggvény. A mozgástervezés eredménye. Kizárólag ütközésmentes állapotokat tartalmaz.

*Két fő komponens:*

* Pályatervezés
  + A mozgás során felveendő állapotok, és azok sorrendje (térgörbe)
* Időparaméterezés
  + A pálya pontjaihoz társított időparaméterek

Kvázistatikus állapotok: a pályája mentén a robot (elvben) bármikor megállítható, és onnan tetszőleges idő után újra indítható

*Kvázistatikus állapot leírása:*

* a mozgási szabadságfokok koordinátái (robotnál: csuklókonfiguráció)
* Kofiguráció: Olyan n-dimenziós vektor, amely egyértelműen jellemzi a robot statikus állapotát

Konfigurációs tér: A robot lehetséges konfigurációs pontjainak halmaza

Kétszabadságfokú robotkar munka és konfigurációs tere

Munkatér 
Konfigurációs tér 
cobs 

### Jelölések, definíciók:

* **W - Munkatér (workspace)** -- a mozgás lehetséges tere (x,y,z koordinátákkal jellemezve)
* **C - Configuration Space** -- a robot lehetséges konfigurációinak halmaza (csuklókoordinátákkal jellemezve)
* **C\_obs - Set of Configurated Obstacles**
* **C\_free - Free Space**
* **q eleme C - konfiguráció**

Lineáris interpoláció eredménye:

#### 

Munkatér 
Konfigurációs tér 

#### Ütközésmentes pályatervezés:

Munkatér 
Konfigurációs tér 

#### Többdimmenziós konfigurációs tér

Munkatér 
3000 
2500 
2000 
E 1500 
1000 
500 
500 
1000 
2500 
3000 
Konfigurációs tér 
'iii 
1000 
1500 
x [mml 
1500 
x [mm] 
2000 
2000 
1000 
y [mml 

### Mozgástervezési módszerek osztályozása

* Konfigurációs tér dimenziószáma szerint (2,3, többdimenziós)
* Konfigurált akadályok alakja (sokszög, síklapok, tetszőleges alakok)
* Robot alakja (pontszerű, kör, gömb, tetszőleges)
* Robot kinematikája (holonomikus, anholonomikus)

### Globális és Lokális mozgástervezés

|  |  |  |
| --- | --- | --- |
| Szempont | **Globális mozgástervezés** | **Lokális akadályelkerülés** |
| *Cél megadása* | Teljes konfiguráció | Csak pozíció |
| *Környezet* | Ismert | Ismeretlen, vagy változó |
| *Térkép* | Globális, pontos térkép | Nincs térkép, vagy csak lokális mérések |
| *Robot alakja* | Pontos modell | Általában körnek tekintik |
| *Tervezés módja* | Tervezés és végrehajtás külön fázisban | Tervezés és végrehajtás nem különül el (helyben történik) |
| *Megoldás jellege* | Globális megoldás garantálható | Csapda-szituációkra érzékeny |
| *Számításigény* | Magasabb (off-line) | Alacsonyabb (on-line) |

## Akadályelkerülési módszerek

#### Működési elvük alapelvei:

* Adott periódusonként lefut az algoritmus
  + Ennek a periódusidőnek annyinak kell lennie, hogy két mintavétel között a robot elmozdulásának kisebbnek kell lennie, mint a szenzorok érzékelési távolsága
* Minden ciklusban a cél egy olyan beavatkozó jel generálása, amely nem okoz ütközést, és közelebb viszi a robotot a céljához

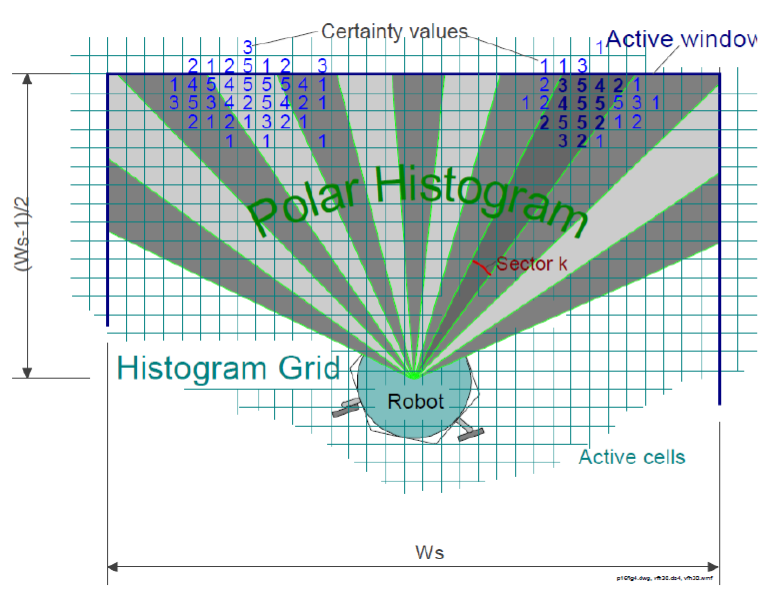
### Virtuális erőtér

* **VFF -** Virtual Force Field
* *Alapötlet*: a robot mozgásának meghatározására virtuális erőket vezet be. Az akadályok taszító a célpont pedig vonzó hatást fejtenek ki a robotra
* Hisztogram rács:
  + Lokális foglaltsági térkép
  + Minden cellához foglaltsági érték van rendelve
  + A rács értékei menet közben folyamatosan frissülnek
  + A legkisebb ellenállás irányába megy a robot
* ***Lokális minimum csapdák****:* a robot olyan minimum pontba fut, ahol a cél felé vezető út le van zárva (pl. U alakú akadály).
  + *Megoldás*: különböző heurisztikákkal, globális tervezés bevonásával
* ***Közeli akadályok közötti áthaladás meghiúsulása:*** a közeli akadályok (pl. ajtó két oldala) miatt létrejövő taszító erő éppen ellentétes irányba mutat, és legyőzi a célpont vonzó erejét 🡪 nem tud áthaladni a robot
* ***Oszcilláció keskeny folyosók esetén***: keskeny folyosón való áthaladás esetén csak a középvonalon való haladás az "energiaminimum". Ha valami miatt kitér egyik irányba, rögtön taszító erő lép fel, ami átlökheti a túloldalra 🡪 ide-oda cikkcakk

### VFH Vector-Field-Histogram

#### Adatredukció:

* A kétdimenziós hisztogram rács aktív ablakának foglaltsági információját egydimenziós poláris hisztogrammá transzformálja (ezzel reprezentálva a poláris akadálysűrűséget)
* A poláris hisztogram alappján történik a kívánt haladási irány és sebesség meghatározása



### Dinamikus ablak módszer (olyan, mintha egér billentyűzetről controllerre váltanánk)

* *DWA* - Dynamic Window Approach
* ***Dinamikus ablak***: merre mehet a robot.

actual 
velocity 
dynamic 
window 
6.0 

* Méretét meghatározza
  + Gyorsuláskorlátok
  + Algoritmus ciklusideje
* Mindig a gyorsulásablakból kell választani a sebességeket, és akkor megvalósítható mozgást kapunk

#### DWA algoritmus

* Keresési tér szűkítése
  + Körív alakú trajektória elemek
    - Csak körívek sorozatából álló trajektóriákat veszünk figyelembe
    - A trajektória elemeket egyértelműen meghatározzák a hozzájuk tartozó (v, omega) sebességek
  + Dinamikus ablak
    - Csak a következő időszeletben elérhető sebességeket vesszük figyelembe
  + Megengedhető sebességek
    - Csak az ütközésmentes haladást lehetővé tévő sebességeket vesszük figyelembe
    - Ha kell, megállást
* Optimalizálás
  + a fennmaradó sebességek közül azt választjuk ki, amelyik maximálja a célfüggvényt:

C:\Users\asarpi\AppData\Local\Temp\msohtmlclip1\02\clip_image003.png

* + - *Heading*: értéke akkor a legnagyobb, ha a robot pontosan a cél felé mozog

C:\Users\asarpi\AppData\Local\Temp\msohtmlclip1\02\clip_image004.png

* + - *Distance*: a köríven levő pályán a legközelebbi akadály körív menti távolságát adja vissza. Minél nagyobb, annál jobb
    - *Velocity*: egész egyszerűen a legnagyobb haladási sebességet jutalmazza
* ***Lokális minimum csapdák:*** az algoritmus nem vesz tudomást a szabad tér összefüggéseiről --> érzékeny a lokális minimum csapda-szituációkra
* ***Súlyok megválasztása***: a célfüggvényben szereplő súlyokat empírikusan kell behangolni. Különböző szituációk különböző súlyokat igényelnek