

Sigur! Iată varianta completă a documentației, inclusiv introducerea nouă și toate detaliile despre algoritmi:

Algoritmi utilizați în controlul robotic - Documentație Detaliată

1. Introducere

În domeniul roboticii, algoritmi joacă un rol esențial în controlul mișcărilor, optimizarea traseelor și evitarea obstacolelor. Această documentație explorează o selecție de algoritmi fundamentali utilizați în proiectul **RoboticControl4Objects**, împărțiți în patru categorii principale:

- **Algoritmi de control optimizat** – responsabili pentru reglarea precisă a mișcărilor robotului și menținerea stabilității acestuia.
- **Algoritmi de planificare a mișcării** – folosiți pentru a calcula traseul optim pe care robotul trebuie să îl urmeze.
- **Algoritmi de evitare a obstacolelor** – esențiali pentru navigația autonomă, prevenind coliziunile.
- **Algoritmi euristici și de învățare automată** – implementați pentru optimizare și adaptare în medii dinamice.

Această documentație detaliază modul în care acești algoritmi funcționează, avantajele fiecăruia și aplicațiile lor practice în robotică.

2. Algoritmi de Control Optimizat

Acești algoritmi sunt folosiți pentru ajustarea și optimizarea mișcărilor unui robot. Controlul optimizat asigură precizie, stabilitate și eficiență în operarea robotică.

2.1 PID (Proportional-Integral-Derivative)

- **Descriere:** Controlerul PID este unul dintre cei mai utilizați algoritmi în controlul robotic datorită capacității sale de a ajusta dinamic mișcările și de a corecta erorile sistemului.
- **Componente:**
 - **Proportional (P):** Ajustează ieșirea proporțional cu eroarea curentă.
 - **Integral (I):** Compensează erorile acumulate în timp.
 - **Derivative (D):** Prezice tendințele viitoare și ajustează corespunzător.

- **Formula matematică:** $u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt}$ $u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt}$
- **Avantaje:**
 - Reduce oscilațiile și instabilitatea sistemului.
 - Aplicabil într-o varietate de sisteme robotice.
- **Resurse utile:**
 - [PID Algorithm Explained](#)
 - [PID Control in Robotics](#)

2.2 LQR (Linear Quadratic Regulator)

- **Descriere:** Algoritm de control optimizat care minimizează o funcție de cost pentru sistemele dinamice. Este utilizat în special în brațele robotice și în controlul optimizat al roboților mobili.
- **Principiul de funcționare:**
 - Utilizează un model matematic al sistemului pentru a calcula o comandă optimă care minimizează costurile asociate erorii și energiei consumate.
- **Resurse utile:**
 - [LQR Control - Introduction](#)

2.3 MPC (Model Predictive Control)

- **Descriere:** Control predictiv bazat pe model este o metodă de control care folosește un model matematic pentru a prezice viitoarele stări ale sistemului și pentru a calcula acțiuni optime.
 - **Aplicații:**
 - Controlul dronelor și al mașinilor autonome.
 - Roboți industriali care trebuie să anticipeze mișcările viitoare.
 - **Resurse utile:**
 - [MPC in Robotics](#)
-

3. Algoritmi de Planificare a Mișcării

Acești algoritmi ajută robotul să găsească traseul optim într-un mediu dat. Sunt esențiali pentru navigația autonomă.

3.1 Algoritmul A*

- **Descriere:** Algoritm euristic pentru găsirea traseului optim. Utilizează o combinație de căutare deterministă și euristici pentru a găsi cel mai scurt drum între două puncte.
- **Avantaje:**
 - Eficient și optimizat pentru multe tipuri de probleme de navigație.
- **Complexitate:** $O(n \log n)$
- **Resurse utile:**

- [A* Search Algorithm - Computerphile](#)

3.2 Algoritmul lui Dijkstra

- **Descriere:** Algoritm pentru găsirea celui mai scurt drum într-un graf. Funcționează bine pentru grafuri neponderate.
 - **Complexitate:** $O(V^2)$ sau $O(E + V \log V)$ cu heap Fibonacci.
 - **Resurse utile:**
 - [Dijkstra's Algorithm](#)
-

4. Algoritmi de Evitare a Obstacolelor

Acești algoritmi permit robotului să navigheze evitând coliziunile.

4.1 Câmpuri Potențiale Artificiale (APF)

- **Descriere:** Algoritm care simulează atracția către obiectiv și respingerea obstacolelor. Robotul este ghidat ca și cum ar fi influențat de un câmp de forțe virtuale.
- **Avantaje:**
 - Metodă rapidă și eficientă pentru evitarea obstacolelor.
- **Resurse utile:**
 - [Artificial Potential Fields - Robotics](#)

4.2 Dynamic Window Approach (DWA)

- **Descriere:** Algoritm utilizat în navigația roboților mobili pentru a calcula viteza optimă evitând coliziunile iminente.
 - **Resurse utile:**
 - [DWA Path Planning](#)
-

5. Algoritmi Euristici și de Învățare Automată

Acești algoritmi sunt folosiți pentru optimizare și luarea deciziilor bazate pe date istorice.

5.1 Algoritmi Genetici (GA - Genetic Algorithm)

- **Descriere:** Algoritm inspirat din selecția naturală pentru optimizarea soluțiilor. Este utilizat în robotică pentru optimizarea traseelor și parametrilor sistemelor complexe.
- **Resurse utile:**
 - [Genetic Algorithms Explained](#)

5.2 Optimizarea prin Roi de Particule (PSO - Particle Swarm Optimization)

- **Descriere:** Algoritm inspirat din comportamentul roiurilor de păsări și pești, utilizat pentru optimizare globală.
 - **Resurse utile:**
 - [PSO Algorithm - Introduction](#)
-

6. Concluzii

Această documentație oferă o bază solidă pentru înțelegerea algoritmilor utilizați în controlul robotic. Pentru detalii suplimentare, consultă *Springer Handbook of Robotics*.
