

NAMA : MUHAMMAD NUR MAJIID
FAKULTAS : STEI-K
NIM : 19624192

1. Algoritma kinematika adalah algoritma yang digunakan untuk menghitung posisi dan orientasi end-effector robot.

algoritma kinematika pada robot meliputi:

Forward kinematics, yaitu Algoritma dasar yang menghitung posisi dan orientasi end-effector robot berdasarkan panjang tautan dan sudut sambungannya.

Inverse kinematics, yaitu Penggunaan persamaan kinematik untuk menentukan gerak robot agar mencapai posisi yang diinginkan.

2. Active disturbance rejection control (atau ADRC, juga dikenal automatic disturbance rejection control) ADRC merupakan pengembangan metode kendali yang berasal dari metode kendali close loop PID.

Perbedaan secara signifikan pada metode kendali close loop PID dan ADRC adalah pada ADRC terdapat pada bagian blok ADRC yang terdiri dari extended state observer (ESO), tracking differentiator (TD), dan nonlinear state error feedback (NLSEF) yang mana ketiga tersebut menerima inputan berupa feedback dari proses pada sistem. kemudian keluaran dari ADRC akan dikirim lagi ke kontroler, sehingga kontroler akan melakukan aksi dengan tujuan agar proses dapat berjalan dengan optimal.

Untuk desain autopilot, strategi ADRC memungkinkan pelacakan yang tepat untuk perintah referensi kecepatan dan ketinggian saat terjadi gangguan parametrik dan gangguan atmosfer yang parah dengan hanya menggunakan informasi keluaran yang terukur. Gangguan terkonsentrasi total dalam sistem menggunakan pengamat yang diperluas untuk observasi dan eliminasi. Dengan cara ini, kemampuan antgangguan sistem ditingkatkan secara efektif.

3. Algoritma PID (Proporsional, Integral, dan Derivatif) adalah algoritma kontrol yang digunakan untuk mengendalikan berbagai proses, seperti suhu, aliran, tekanan, dan kecepatan. Algoritma ini bekerja dengan menghitung nilai kesalahan sebagai perbedaan antara variabel proses yang diukur dan titik setel yang diinginkan.

Kemudian, pengontrol akan berusaha meminimalkan kesalahan tersebut dengan menyesuaikan variabel kontrol.

4. Algoritma A^* (A Star) adalah algoritma pencarian yang digunakan untuk menemukan jalur terpendek antara titik awal dan akhir. Algoritma ini sering digunakan untuk penjelajahan peta guna menemukan jalur terpendek yang akan diambil. A^* awalnya dirancang sebagai masalah penjelajahan graph (graph traversal), untuk membantu robot agar dapat menemukan arahnya sendiri.

A^* saat ini masih tetap menjadi algoritma yang sangat populer untuk graph traversal. Algoritma A^* mencari jalur yang lebih pendek terlebih dahulu, sehingga menjadikannya algoritma yang optimal dan lengkap. Algoritma yang optimal akan menemukan hasil yang paling murah dalam hal biaya untuk suatu masalah, sedangkan algoritma yang lengkap menemukan semua hasil yang mungkin dari suatu masalah. Aspek lain yang membuat A^* begitu powerful adalah penggunaan graph berbobot dalam penerapannya.

Graph berbobot menggunakan angka untuk mewakili biaya pengambilan setiap jalur atau tindakan. Ini berarti bahwa algoritma dapat mengambil jalur dengan biaya paling sedikit, dan menemukan rute terbaik dari segi jarak dan waktu. Adapun kelemahan utama dari algoritma ini adalah kompleksitas ruang dan waktunya. Algoritma A^* membutuhkan banyak ruang untuk menyimpan semua kemungkinan jalur dan banyak waktu untuk menemukannya.