

Kinematics

Dalam fisika, kinematika mendeskripsikan gerak benda tanpa menghiraukan penyebab gerak tersebut (dinamika). Kinematika sangat terpakai dalam bidang robotika untuk membuat identifikasi objek dengan metode pengklasifikasian dan pembedaan tiap objek menggunakan frame sebagai batasan spasial (object detection), menentukan solusi optimal (karena bisa ada banyak variasi solusi) dari kombinasi pergerakan guna mencapai posisi tertentu (pose estimation), dan pemrosesan visual (mapping gambar 2d (input) pada ruang 3d (output)). Ini terpakai di motion detection, legan robot, dan kamera di mobil tanpa pengemudi.

ADRC

ADRC dapat dipandang sebagai metode adaptasi kontinu terhadap faktor-faktor lingkungan yang tidak terduga dan ternyata mempengaruhi performa robot. ADRC dimulai dengan mengambil input berupa data lingkungan, lalu menghasilkan output berupa feedback control sebagai strategi dalam merespons situasi tersebut. Setelah itu, dilakukan evaluasi berdasarkan performa sesudahnya. Evaluasi tersebut turut menjadi masukan/input bagi output selanjutnya. Namun, tidak semua situasi menjadi input, karena bisa jadi hanyalah disturbance sementara yang jika ditanggapi malah menurunkan performa robot secara keseluruhan.

Kelebihan ADRC adalah tidak memerlukan pemodelan matematis yang presisi karena naturnya yang terus beradaptasi secara menjadi semakin baik (bahasa kerennya: nonlinear). Akan dibahas bahwa hal ini menjadi pembeda utama ADRC dengan PID.

Salah satu pengaplikasian konkret dari ini adalah di UAV di mana perlu beradaptasi dengan kecepatan angin yang sangat tidak menentu. Contoh lainnya adalah robot yang dikendalikan dari jarak jauh yang perlu beradaptasi dengan lingkungan guna meminimalisir menabrak benda berbahaya (object avoidance).

PID

PID memiliki kemiripan dengan ADRC di mana keduanya beradaptasi secara kontinu. Perbedaananya terletak di mekanisme adaptasi/tuning keduanya. Jika ADRC lebih abstrak/fleksibel, PID lebih matematis dan sangat memakai kalkulus (seperti kepanjangannya, yaitu proportional, integral, and derivative). Oleh sebab itu, ketika menggunakan PID diperlukan pemahaman yang menyeluruh tentang pemodelan matematis sistem karena parameter tuning-nya dibatasi berdasarkan model tersebut untuk menghasilkan model ter-update (sehingga bisa prediksi strategi di depan juga berdasarkan pemodelan ini). Untuk penjelasan lebih teknisnya: proporsional = error sekarang sebagai input, integral = error akumulatif sebagai input yang diproses, dan derivatif = memprediksi error selanjutnya berdasarkan laju perubahan sebagai output.

A*

A* adalah salah satu search algorithm yang bertujuan mencari “jalur dengan biaya termurah” dari satu node/lokasi data ke node lain. Sebetulnya tidak perlu dalam konteks data structure, bisa juga tentang hal-hal lain seperti jalan terdekat dari kos saya menuju kantin GKU 2 ITB, pergerakan zombie di Resident Evil dalam mengejar saya (supaya cepat ngejanya), maupun robot dalam bergerak dengan efisien sambil menghindari banyak obstacles/penghalang.

A* sendiri tidak memiliki detail operasional yang absolut karena bergantung pada fungsi heuristiknya. A* bersifat heuristik yang berarti memakai metode trial and error (coba satu-satu, tetapi ada aturannya supaya optimal berdasarkan fungsi heuristik tersebut). Coba-cobanya dibuat sedemikian rupa sehingga mencoba jalur dengan cost terkecil dahulu, baru cost yang lebih besar jika masih belum bisa sampai pada tujuannya. Metode trial and error ini bisa merugikan jika fungsi heuristiknya kurang efisien atau jika dataset nya sangat besar.