

Title: Machine learning control

URL: https://en.wikipedia.org/wiki/Machine_learning_control

PageID: 53802271

Categories: Category:Control theory, Category:Cybernetics, Category:Machine learning

Source: Wikipedia (CC BY-SA 4.0).

Machine learning control (MLC) is a subfield of machine learning , intelligent control , and control theory which aims to solve optimal control problems with machine learning methods. Key applications are complex nonlinear systems for which linear control theory methods are not applicable.

Types of problems and tasks

Four types of problems are commonly encountered:

Control parameter identification: MLC translates to a parameter identification [1] if the structure of the control law is given but the parameters are unknown. One example is the genetic algorithm for optimizing coefficients of a PID controller [2] or discrete-time optimal control. [3]

Control design as regression problem of the first kind: MLC approximates a general nonlinear mapping from sensor signals to actuation commands, if the sensor signals and the optimal actuation command are known for every state. One example is the computation of sensor feedback from a known full state feedback . Neural networks are commonly used for such tasks. [4]

Control design as regression problem of the second kind: MLC may also identify arbitrary nonlinear control laws which minimize the cost function of the plant. In this case, neither a model, the control law structure, nor the optimizing actuation command needs to be known. The optimization is only based on the control performance (cost function) as measured in the plant. Genetic programming is a powerful regression technique for this purpose. [5]

Reinforcement learning control: The control law may be continually updated over measured performance changes (rewards) using reinforcement learning . [6] [7]

Adaptive Dynamic Programming

Adaptive Dynamic Programming (ADP), also known as approximate dynamic programming or neuro-dynamic programming, is a machine learning control method that combines reinforcement learning with dynamic programming to solve optimal control problems for complex systems. ADP addresses the " curse of dimensionality " in traditional dynamic programming by approximating value functions or control policies using parametric structures such as neural networks. The core idea revolves around learning a control policy that minimizes a long-term cost function J $\{ \displaystyle J \}$, defined as $J (x (t)) = \int t \infty e - \gamma (\tau - t) r (x (\tau) , u (\tau)) d \tau$ $\{ \displaystyle J(x(t)) = \int_{-\infty}^t e^{-\gamma(\tau - t)} r(x(\tau), u(\tau)) d\tau \}$, where x $\{ \displaystyle x \}$ is the system state, u $\{ \displaystyle u \}$ is the control input, r $\{ \displaystyle r \}$ is the instantaneous reward, and γ $\{ \displaystyle \gamma \}$ is a discount factor. ADP employs two interacting components: a critic that estimates the value function $V (x) \approx J (x)$ $\{ \displaystyle V(x) \approx J(x) \}$, and an actor that updates the control policy $u (x)$ $\{ \displaystyle u(x) \}$. The critic and actor are trained iteratively using temporal difference learning or gradient descent to satisfy the Hamilton-Jacobi-Bellman (HJB) equation :

$$\min_u (r (x , u) + \partial V / \partial x f (x , u)) = 0 , \{ \displaystyle \min_u \left(r(x,u) + \frac{\partial V}{\partial x} f(x,u) \right) = 0 , \}$$

where $f (x , u)$ $\{ \displaystyle f(x,u) \}$ describes the system dynamics. Key variants include heuristic dynamic programming (HDP), dual heuristic programming (DHP), and globalized dual heuristic programming (GDHP). [7]

ADP has been applied to robotics, power systems, and autonomous vehicles, offering a data-driven framework for near-optimal control without requiring full system models. Challenges remain in ensuring stability guarantees and convergence for general nonlinear systems.

Applications

MLC has been successfully applied

to many nonlinear control problems,

exploring unknown and often unexpected actuation mechanisms. Example applications include:

spacecraft attitude control , [8]

thermal control of buildings, [9]

feedback control of turbulence , [2] [10]

and remotely operated underwater vehicles . [11]

Many more engineering MLC application are summarized in the review article of PJ Fleming & RC Purshouse (2002). [12]

As is the case for all general nonlinear methods,

MLC does not guarantee convergence, optimality , or robustness for a range of operating conditions.

See also

Reinforcement learning

References

Further reading

Dimitris C Dracopoulos (August 1997) "Evolutionary Learning Algorithms for Neural Adaptive Control" , Springer. ISBN 978-3-540-76161-7 .

Thomas Duriez, Steven L. Brunton & Bernd R. Noack (November 2016) "Machine Learning Control - Taming Nonlinear Dynamics and Turbulence" , Springer. ISBN 978-3-319-40624-4 .