Математические Методы Распознавания образов, весна, ВМК МГУ
Семинар №10-11

Автор: Афанасьев Глеб

1 Введение

Данный блок из трех семинаров является развитием такой области как оптимальное управление в сторону применения алгоритмов машинного обучения.

В некотором смысле, он также является продолжением блока по оптимизации, поэтому, рекомендуется ознакомиться с ним перед освоением данного материала, несмотря на то что основные моменты будут напомнены.

2 Лекция 1: задача оптимального управления и методы ее решения

§2.1 Постановка задачи оптимизации, условия ККТ

На предыдущих семинарах вами была рассмотрена задача оптимизации. Напомним один из вариантов ее математических постановок:

$$f(x) \to min_x$$
$$g_i(x) \leqslant 0, i = 1, ..., m$$
$$g_i(x) = 0, i = m+1, ..., s$$

Также, были рассмотрены основные подходы к ее решению. Одним из важных теоретических результатов являются необходимые условия локального экстремума, или же теорема Каруша-Куна-Таккера:

Если точка x^* - точка локального минимума в задаче оптимизации, тогда найдутся такие $\lambda^* = (\lambda_0^*, ..., \lambda_s^*)$ такие, что:

$$\lambda^* \neq 0 \tag{2.1}$$
$$\lambda_i^* \geqslant 0, i = 1, ..., m \tag{2.2}$$
$$\lambda_i^* g_i(x) = 0, i = 1, ..., s \tag{2.3}$$
$$\nabla_x f(x) + \lambda \nabla_x g_i(x) = 0, i = 1, 2, ..., s \tag{2.4}$$

Однако, данный результат является аналитическим, реализовать который для реальных задач в большинстве случаев невозможно. На практике обычно мы хотим получить итерационный алгоритм, сходящийся к оптимальной точке. Для того чтобы лучше понять логику дальнейших рассуждений, напомним еще одну интерпретацию условий ККТ.

Заметим что задача оптимизации в указанном выше виде эквивалентна следующей задаче:

$$min_x max_{\lambda \geqslant 0}[f(x) + \lambda_1 g_1(x) + \lambda_2 g_2(x) + ... + \lambda_s g_s(x)]$$

При $g_i(x) > 0$ внутренняя задача максимизации уходит в бесконечность, тогда как для $g_i(x) \leqslant 0$ внешняя задача минимизации имеет смысл. Заметим что выражение, стоящее в квадратных скобках является в точности лагранжианом исходной задачи.

Таким образом, мы получаем что условия ККТ определяют седловую точку для лагранжиана при условиях на λ.

§2.2 Постановка задачи оптимального управления, ККТ для нее

Теперь перейдем к целевой задаче данного блока: задаче оптимального управления. В общем случае, задача заключается в поиске управления и порождаемой им траектории управляемого объекта, оптимальных в смысле заданного функционала, и удовлетворяющих ограничениям на динамику. В большинстве статей задача выписывается в следующем виде:

$$J(x(t), u(t)) = \int_{t_0}^{t^1} l(x(t), u(t))dt + l_F(x(t_1)) \to min_{x,u}$$
$$\dot{x} = f(x(t), u(t))$$

Здесь l - функция, определяющая потери, получающиеся если в состоянии x управляемый объект выполнит действие u, l_F - функция, определяющая потери, в конечном состоянии. В таком виде решением задачи является пара функций $x^*(t), u^*(t)$, где $x^*(t)$ - оптимальная траектория, $u^*(t)$ - порождающее ее оптимальное управление.

На практике мы чаще имеем дело с дискретным вариантом данной задачи:

$$J(x(t), u(t)) = \sum_{n=1}^{N-1} l(x_n, u_n) + l_F(x_N) \to min_{x_{1:N}, u_{1:N-1}}$$
$$x_{n+1} = f(x_n, u_n)$$

И задача таким образом сводится к поиску оптимальной траектории вида $((x_1, u_1), (x_2, u_2), ..., (x_{N-1}, u_{N-1}), (x_N))$, минимизирующей функционал J и удовлетворяющей ограничениям на динамику. Важно помнить, что дискретными тут являются траектория и подсчет функционала, в то время как функции потерь l, l_F и f могут быть непрерывными.

То есть, мы имеем оптимизационную задачу с ограничением типа "равенство".

Попробуем найти необходимое условие оптимальности по теореме ККТ. Выпишем лагранжиан:

$$L = \sum_{n=1}^{N-1} [l(x_n, u_n) + \lambda_{n+1}^T (f(x_n, u_n) - x_{n+1})] + l_F(x_N)$$

Теперь предположим что у нас есть оптимальное решение x^*, u^* попробуем выписать условия ККТ:

$$\sum_{n=1}^{N-1} \frac{\partial l(x_n^*, u_n^*)}{\partial x} + {\lambda_{n+1}^*}^T \frac{\partial (f(x_n^*, u_n^*) - x_{n+1}^*)}{\partial x} + \frac{\partial l_F(x_N^*)}{\partial x} = 0 \tag{2.5}$$

$$\lambda_n^* \geqslant 0, n = 1, ..., m \tag{2.6}$$

$${\lambda_{n+1}^*}^T (f(x_n^*, u_n^*) - x_{n+1}^*), n = 2, ..., N \tag{2.7}$$

$$\lambda^* \neq 0 \tag{2.8}$$

Заметим, во-первых, что у нас задача имеет N-1 ограничений. Следовательно, двойственных переменных λ также будет N-1. Поэтому, для удобства дальнейших выкладок нумерацию двойственных переменных мы будем начинать с 2. То есть, λ_1 не существует и эта переменная нигде не используется. Во-вторых, вспомним, что условие (2.6) ставилось только для ограничений типа неравенства, поэтому в нашем случае, оно не имеет смысла. В-третьих, очевидно что условие (2.7) будет выполняться для решения задачи всегда, так как мы имеем дело только с ограничениями типа равенство. Таким образом, условия ККТ могут быть переписаны в виде:

$$\sum_{n=1}^{N-1} \frac{\partial l(x_n^*, u_n^*)}{\partial x} + {\lambda_{n+1}^*}^T \frac{\partial (f(x_n^*, u_n^*) - x_{n+1}^*)}{\partial x} + \frac{\partial l_F(x_N^*)}{\partial x} = 0 \tag{2.9}$$

$$x_{n+1}^* = f(x_n^*, u_n^*), n = 1, ..., N \tag{2.10}$$

$$\lambda^* \neq 0 \tag{2.11}$$

Где (2.10) означает лишь то что решение должно удовлетворять исходным ограничениям на динамику системы.

§2.3 Принцип Максимума Понтрягина

Теперь вспомним физику. Для динамических систем в курсе физики вводилось понятие функции Гамильтона $H(q, p)$ - функции от двух равномощных групп переменных (путь мощность равна N), каждая из которых связана с функцией Гамильтона следующим образом:

$$\dot{q_i} = \frac{\partial H(q, p)}{\partial p_i}, i = 1, ..., N \tag{2.12}$$

$$\dot{p_i} = -\frac{\partial H(q, p)}{\partial q_i}, i = 1, ..., N \tag{2.13}$$

И при этом функция константна по времени:

$$\frac{\partial H(q(t), p(t))}{\partial t} = 0$$

Из функции Гамильтона выводилось множество интересных свойств динамических систем. Попробуем ввести аналогичную функцию для нашей системы следующим образом:

$$H(x, \lambda, u) = \langle \lambda, f(x, u) \rangle + l(x, u)$$

Сначала выразим функцию Лагранжа через функцию Гамильтона:

$$L(x, \lambda, u) = H(x_1, \lambda_2, u_1) + \sum_{n=2}^{N-1} [H(x_n, \lambda_{n+1}, u_n) - \lambda_n^T x_n] + l_F(x_N) - \lambda_N^T x_N$$

$$\frac{\partial L(x, \lambda, u)}{\partial x_n} = \frac{\partial H(x, \lambda, u)}{\partial x_n} - \lambda_n = 0$$

$$\frac{\partial L(x, \lambda, u)}{\partial \lambda_n} = \frac{\partial H(x, \lambda, u)}{\partial \lambda_n} - x_{n+1} = 0$$

То есть:

$$x_{n+1} = \nabla_{\lambda_n} H(x_n, \lambda_{n+1}, u_n) \tag{2.14}$$
$$\lambda_n = \nabla_{x_n} H(x_n, \lambda_{n+1}, u_n) \tag{2.15}$$
$$\lambda_N = \frac{\partial l(x_N)}{\partial x_N} \tag{2.16}$$

Выписанные условия являются уравнениями Гамильтона в дискретном случае. Для задачи оптимального управления, они также являются подмножеством условий принципа максимума Понтрягина, изученного вами на курсе по оптимальному управлению. ПМП гласит что если управление u оптимально то оно удовлетворяет условиям (2.14 - 2.16) и является экстремумом функции Гамильтона. То есть:

$$\frac{\partial H(x_{n,n+1}, u_n)}{u_n} = 0$$

Здесь мы видим важный результат: выражение x_{n+1} через x_n с соблюдением условий минимизации. Попробуем применить этот результат к модельной задаче.

$$\sum_{n=1}^{N-1} \frac{1}{2} [x_n^T Q_n x_n + u_n^T R_n u_n + x_N^T \theta X_N] \to min_{x,u}$$

$$x_{n+1} = A_n x_n + B_n u_n$$

$$Q \geqslant 0, R > 0$$

В теории управления такая система называется линейно квадратичный регулятор (Linear quadratic regulator, LQR)

Выпишем функцию Гамильтона:

$$H = \langle \lambda_{n+1}, A_n x_n + B_n u_n \rangle + l(x, u)$$

Теперь, основываясь на принципе максимума Понтрягина, мы можем записать следующий алгоритм:

1. Инициализируем случайно $u_{1:N}$
2. Сгенерируем траекторию $x_{1:N}$ через динамику: $x_{n+1} = A_n x_n + B_n u_n$
3. Генерируем сопряженные переменные $\lambda_{2:N}$ через условие (2.15) ПМП: $\lambda_n = Q_n x_n + A_n^T \lambda_{n+1}$
4. Генерируем последнюю сопряженную переменную (2.16) $\lambda_N = Q_N x_N$
5. Выводим новое управление как минимизатор функции Гамильтона: $u_n = argmax_u H(x_n, \lambda_{n+1}, u_n) = -R^{-1} B^T \lambda_{n+1}$
6. Повторяем шаги 2-5 пока управление не перестанет меняться

Вообще говоря, данный алгоритм применим не только к рассмотренной задаче, но и ко всем задачам подобного вида.

§2.4 Уравнения Рикатти

Выведем еще один важный результат теории оптимального управления для LQR.

Предположим что нам известно начальное условие $x_1 = \overline{x}$

Введем вектор $z : z = [u_1, x_2, u_2, x_3, ..., x_N]^T$

И матрицу:

$$H_{2N-2\times 2N-2} = \begin{bmatrix} R_1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & Q_2 & 0 & 0 & \cdots & 0 \\ 0 & 0 & R_2 & 0 & \cdots & 0 \\ 0 & 0 & 0 & Q_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & Q_N \end{bmatrix}$$

Теперь мы можем переписать оптимизируемый функционал:

$$J = \frac{1}{2} z^T H z$$

Введем еще два обозначения:

$$C = \begin{bmatrix} \theta & -I & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & A & B & -I & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & A & B & -I & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & A & B & -I \end{bmatrix}$$

$$d = [-Ax_1, 0, 0, 0, ...0]^T$$

Таким образом, задача может быть переписана в следующем виде:

$$\frac{1}{2}z^T Hz \to min_z$$
$$Cz = d$$

Лагранжиан будет выглядеть следующим образом:

$$L(z, \lambda) = \frac{1}{2}z^T Hz + \lambda^T (Cz - d)$$

Его градиенты:

$$\nabla_z L(z, \lambda) = Hz + c^T \lambda = 0$$

$$\nabla_\lambda L(z, \lambda) = Cz - d = 0$$

Таким образом, мы можем получить седловую точку как решение СЛАУ:

$$\begin{bmatrix} H & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} z \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ d \end{bmatrix}$$

Для выведения общей схемы решения рассмотрим пример для векторов размерности 4:

$$\begin{bmatrix} R & 0 & 0 & 0 & 0 & 0 & B^T & 0 & 0 \\ 0 & Q & 0 & 0 & 0 & 0 & -I & A^T & 0 \\ 0 & 0 & R & 0 & 0 & 0 & 0 & B^T & 0 \\ 0 & 0 & 0 & Q & 0 & 0 & 0 & -I & A^T \\ 0 & 0 & 0 & 0 & R & 0 & 0 & 0 & B^T \\ 0 & 0 & 0 & 0 & 0 & Q & 0 & 0 & -I \\ B & -I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & A & B & -I & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & A & B & -I & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ x_2 \\ u_2 \\ x_3 \\ u_3 \\ x_4 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -Ax_1 \\ 0 \\ 0 \end{bmatrix}$$

Получаем следующую схему решения:

$$Qx_4 - \lambda_4 = 0 \Longrightarrow \lambda_4 = Qx_4$$
$$Ru_3 + B^T \lambda_4 = Ru_3 + B^T Qx_4 = 0 \Longrightarrow Ru_3 = -B^T Q(Ax_3 + Bu_3) \Longrightarrow$$
$$\Longrightarrow u_3 = -(R + B^T QB)^{-1} B^T QAx_3$$

Обозначим

$$K = (R + B^T QB)^{-1} B^T QA$$

$$Qx_3 - \lambda_3 + A^T \lambda_4 = 0$$

$$\lambda_3 = Qx_3 + A^T Q(Ax_3 + Bu_3) = Qx_3 + A^T Q(Ax_3 + Bu_3) = (Q + A^T Q(A - BK))x_3$$

Таким образом, мы получаем процесс восстановления траектории и управления вычисляя их в общем случае для различных Q и R проходом вперед и назад по следующему алгоритму:

$$\begin{cases} P_N = Q_N \\ K_n = (R + B^T P_{n+1} B)^{-1} B^T P_{n+1} A \\ P_n = Q + A^T P_{n+1}(A - BK_n) \\ u_n = K_n x_n \\ x_n = A_{n-1} x_{n-1} + B_{n-1} u_{n-1} \\ \lambda_n = P_n x_n \end{cases}$$

Данные уравнения называются уравнениями Рикатти. В более общем случае вы могли встречать их на курсе по оптимальному управлению.

3 Лекция 2: уравнение Беллмана, связь с Reinforcement Learning

Мы рассмотрели схему решения задач оптимального управления путем нахождения полной траектории в виде вектора фиксированной длины исходя из известной динамики. Кроме того, длина траектории являлась, в некотором роде, гиперпараметром и задавалась явно перед формулировкой задачи. В данной лекции мы рассмотрим способ отойти от указанных ограничений. Для этого подойдем к решению задач ОУ немного с другой стороны. Более подробно вы его рассматривали на одноименных лекциях.

§3.1 Уравнение Беллмана

Рассмотрим общую задачу минимизации:

$$f(x) \to min_{x \in X} \tag{3.1}$$

идея нового подхода заключается в получении задачи ОУ как расширение задачи (3.1) путем добавления в нее нового параметра. Пока обозначим его как $\alpha \in \Lambda$. Теперь новая задача выглядит следующим образом:

$$F(x, \alpha) \to min_{x \in X(\alpha)}$$

Очевидно что при каком-то $\alpha = \alpha_0$ мы приходим к оригинальной задаче:

$$F(x, \alpha_0) = f(x)$$

$$X(\alpha_0) = X$$

Введем функцию следующего вида:

$$T(\alpha) = min_{x \in X(\alpha)} F(x, \alpha), \forall \alpha \in \Lambda$$

Функцией Беллмана называется функция T взятая со знаком минус:

$$V(\alpha) = -T(\alpha)$$

Теперь вспомним что изначально у нас стояла задача минимизации с ограничениями на динамику:

$$J(x(t), u(t)) = \int_{t_0}^{t^1} l(x(t), u(t))dt \to min_{x,u}$$
$$\dot{x} = f(x(t), u(t))$$

Кроме того, зачастую мы хотим задать точки начала и конца траектории: $x(t_0) = x_0, x(t_1) = x_1$

Самый распространенный способ ввести параметризацию данной задачи - через точку начала траектории.

То есть от исходной задачи мы переходим к задаче
$\begin{cases} J(x(t), u(t)) = \int_{t_0}^{t^1} l(x(t), u(t))dt \\ \dot{x} = f(x(t), u(t)) \\ x(t_0) = \alpha \end{cases}$

Конечное условие зачастую задается неявно через оптимизируемый функционал.

Соответственно, для такой задачи функция Беллмана примет следующий вид:

$$V(\alpha) = -min_{x,u \in X_\alpha} J(x, u, \alpha)$$

Для дальнейших рассуждений нам потребуется сделать следующие предположения о функции Беллмана и решаемой задаче:

1. Функция Беллмана существует для всех рассматриваемых α
2. $V(\alpha) \in C^1(\alpha)$
3. Любой "хвост"оптимальной траектории - оптимален

Первые два пункта обычно гарантируются видом конкретной задачи.

Рассмотрим подробнее пункт 3. По сути, он означает что если мы имеем оптимальное управление и траекторию в виде функций от времени $u^*(t), x^*(t)$ для начальной точки $x(t_0) = x_0$ и конечной точки x_1, то это же управление будет оптимальным и для такой же задачи с начальной точкой $x_{0.5}$, где $x_{0.5}$ - точка, лежащая на оптимальной траектории $x^*(t)$.

Для рассматриваемой задачи он проверяется тривиально от противного: зафиксируем точку на траектории $x_{0.5}$ и предположим что из нее существует более оптимальное управление для достижения конечной точки x_1. Обозначим его за $u'(t), x'(t)$ (рис 1). То есть $\int_\tau^{t1} l(x'(t), u'(t))dt < \int_\tau^{t1} l(x^*(t), u^*(t))dt$, где τ - момент времени когда объект оказался в точке $x_{0.5}$.

Тогда совершенно очевидно что управление вида

$$u(t) = \begin{cases} u^*(t), t < \tau \\ u'(t), t \geqslant \tau \end{cases}$$

Будет более оптимальным в смысле оптимизируемого функционала, так как $\int_{t_0}^{\tau} l(x^*(t), u^*(t))dt + \int_\tau^{t1} l(x'(t), u'(t))dt < \int_{t_0}^{t_1} l(x^*(t), u^*(t))dt$. Значит, управление $u^*(t)$ не оптимально и мы приходим к противоречию с условием.

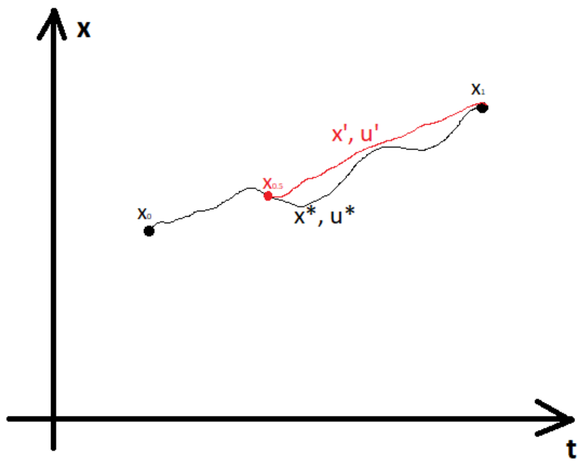

Рис. 1.

Далее, будем считать что предположения 1-3 выполнены.

Теперь рассмотрим оптимальное управление $u(t)$ для задачи выше. Оговоримся что на курсе по оптимальному управлению было принято оперировать оптимальными парами $(x(t), u(t))$, однако поскольку управление порождает траекторию, мы будем в основном ссылаться именно на само управление как на искомый объект, имея ввиду что он порождает соответствующую ему траекторию.

Далее, возьмем промежуток времени Δt: $t_0 + \Delta t < t_1$.

Получаем:

$$T(\alpha) = \int_{t_0}^{t_0+\Delta t} l(x(t), u(t))dt + \int_{t_0+\Delta t}^{t_1} l(x(t), u(t))dt = \int_{t_0}^{t_0+\Delta t} l(x(t), u(t))dt + T(x(t_0+\Delta t))$$

Значит,

$$V(x(t_0 + \Delta t)) - V(\alpha) = L(t_0 + \Delta t) - L(\alpha)$$

где $L(t)$ - первообразная по t для $l(x(t), u(t))$.

Делим правую и левую части на Δt и устремляем Δt к 0. Учитывая что с помощью α мы параметризовали значение $x(t_0)$ получаем:

$$\dot{V}(\alpha) = \dot{L}(t_0) = l(\alpha, u_0)$$

Вспоминаем что α может быть произвольным и должно удовлетворять заданной динамике

$$\dot{V}(\alpha) = V'_x \dot{x}(t) = V'_x f(x, u) = l(x, u)$$

$$V'_x f(x, u) - l(x, u) = 0$$

И в итоге получаем необходимое условие оптимальности:

$$\langle V'_x, f(x, u) \rangle - l(x, u) = 0$$

Теперь предположим что мы сначала на протяжении времени Δt двигаемся каким-то случайным управлением v, а потом включаем оптимальное. Для функции T будет справедливо следующее неравенство:

$$T(\alpha) \leqslant \int_{t_0}^{t_0+\Delta t} l(v(t), x(t))dt + T(y(t_0 + \Delta t))$$

Где $y(t_0+\Delta t)$ - точка, в которой окажется управляемый объект при включении управления v в точке α за время Δt

Перенося функции T в левую часть и деля на Δt получаем:

$$\frac{V(y(t_0 + \Delta t)) - V(\alpha)}{\Delta t} \leqslant \frac{L(t_0 + \Delta t) - L(x_0)}{\Delta t}$$

Раскладывая $y(t_0 + \Delta t)$ в ряд Тейлора, получаем:

$$y(t_0 + \Delta t) = y(t_0) + \dot{y}(t_0)\Delta t + \overline{o}(\Delta t) = \alpha + f(\alpha, v(t_0))\Delta t + \overline{o}(\Delta t)$$

И неравенство можно переписать в виде:

$$\langle V_x', f(x, v(t))\rangle - l(x, v) \leqslant 0 \tag{3.2}$$

То есть, выражение из левой части (3.2) для произвольного управления меньше 0, а для оптимального равно 0. Таким образом, мы получаем уравнение Беллмана:

$$max_{u(t)}\langle V_x', f(x(t), u(t))\rangle - l(x(t), u(t)) = 0$$

§3.2 Связь с ККТ и ПМП

Теперь покажем связь с ККТ и ПМП

Предположим что функция Беллмана - дважды непрерывно дифференцируема. И рассмотрим функцию $\psi(t) = V'(x(t))$.

Тогда

$$\frac{d\psi(t)}{dt} = \frac{d}{dt}V'(x(t)) = V''(x(t))f(x(t), u(t))$$

Для удобства дальнейших выкладок введем функцию $B(x, u) = \langle V_x'(x), f(x, u)\rangle - l(x, u)$

$$B_x'(x, u) = V''(x(t))^T f(x, u) + f_x'(x, u)^T V'(x) - l_x'(x, u)$$

Теперь пусть $u^*(t)$ - оптимальное управление, $x^*(t)$ - порождаемая им траектория. Тогда из уравнения Беллмана совершенно очевидно что $B_x'(x^*, u^*) = 0$, значит

$$V''(x(t))^T f(x, u) = -f_x'(x, u)^T V'(x) + l_x'(x, u)$$

Таким образом,

$$\frac{d\psi(t)}{dt} = V''(x(t))f(x(t), u(t)) = -f_x'(x, u)^T V'(x) + l_x'(x, u) = -f_x'(x, u)^T \psi(t) + l_x'(x, u)$$

Получаем в точности второе условие принципа максимума Понтрягина для случая непрерывной сопряженной переменной и непрерывной функции Гамильтона $H(x, \psi, u) = \langle \psi, f(x, u) \rangle + l(x, u)$

Уравнение Беллмана в таких терминах обозначает что функция Гамильтона на оптимальной траектории выходит на максимум. Только теперь мы еще и знаем его значение.

4 Практическое применение

Теперь разберемся с практическим применением этой теории. Вспомним задачу оптимального управления. Требовалось найти управление u(t), оптимальное по заданному функционалу J. При этом, ставились ограничения на динамику:

$$J(x(t), u(t)) = \int_{t_0}^{t^1} l(x(t), u(t))dt \to min_{x,u}$$

$$\dot{x} = f(x(t), u(t))$$

Так она выглядит в теории. На практике в наше время обычно ее рассматривают в несколько ином виде, накладывая новые условия на вид решения и задачи.

Во-первых, так как расчеты управляющих действий и траектории ведутся на компьютерах, где время дискретно, мы переходим к дискретной постановке задачи. Более того, исторически сложилось что при решении задачи мы хотим не минимизировать потери, а максимизировать награду по всевозможным управлениям. В литературе награда, полученная при выполнении действия u в состоянии x обозначается $r(x, u)$ Таким образом, задача принимает вид:

$$J(x(t), u(t)) = \sum_{n=0}^{N} r(x_n, u_n)dt \to max_u$$

$$x_{n+1} = f(x_n, u_n)$$

Более того, далеко не всегда у нас стоит требование на фиксированное количество шагов, поэтому N часто не фиксировано и в некоторых задачах может равняться бесконечности.

В таких терминах, функция Беллмана $V(x)$ будет равняться максимальной по всевозможным управлениям общей награде при условии что мы начали движение из точки x. Переходную функцию T(x) на практике пропускают.

Ранее мы уже рассмотрели два метода решения задачи такого вида: итерация по ПМП и уравнения Рикатти. Однако, в них мы вычисляли сразу полную траекторию и управление для фиксированной длины, оптимизируя их целиком от 0 до N для поставленной задачи. На практике же нам было бы удобнее получить функцию управления u(.), с помощью которой можно будет управлять агентом в среде произвольное количество времени до достижения требуемой цели. Причем, обычно нас интересует управление, явно зависящее не от времени, а от текущего положения агента, так как в первую очередь он должен принимать решение в зависимости от окружения и своего текущего состояния. И наконец, мы хотим реализовать современные возможности машинного обучения, поэтому представляем управление в виде нейронной сети или иной обучаемой функции. В современной литературе, представленное таким образом управление называется политикой управляемого агента и обозначается $\pi(x|\theta)$, где x - текущее состояние, θ - параметры нейросети/иного алгоритма.

Остается невыясненным вопрос: а как учить эту сеть? Ведь обучающей выборки как в обычном ML/DL у нас нету. Для ответа на него распишем, какой вид примет функция и уравнение Беллмана в рассматриваемой задаче.

$$V(x) = max_{\pi(x|\theta)} \sum_{n=0}^{+\infty} r(x_n, \pi(x_n))$$

Здесь суммирование идет до бесконечности, однако в случае конечных траекторий мы можем просто дополнить нулями оставшиеся награды.

Далее, вспомним как мы выводили уравнение Беллмана. Непрерывность там возникала когда мы переходили к пределу по $\Delta t \to 0$. Распишем вывод для дискретного случая когда на Δt делить не нужно:

$$V(x_k) - V(x_{k+1}) = r(x_k, \pi(x_k|\theta))$$

Так как минимально возможное количество шагов во времени равняется 1 шагу, мы получаем левую часть уравнения. Правая часть следует из того что за 1 шаг мы получим только награду за этот шаг. Вообще говоря, ровно 1 шаг делать не обязательно. Так как изначально требований на Δt не было, мы можем выписать уравнение как

$$V(x_k) - V(x_{k+n}) = \sum_{i=k}^{k+n} r(x_i, \pi(x_i|\theta))$$

Таким образом, мы получили условие, которому должно удовлетворять оптимальное решение и на которое можно пробовать обучать параметры θ.

Собственно говоря, большинство исследований в таком направлении как Обучение с Подкреплением (Reinforcement Learning) направлены на вывод различных видов уравнений Беллмана, поиск способов его продифференцировать по θ и вычислить функцию Беллмана. Некоторые из них, основанные на рассмотренных на прошедших лекциях теоретических достижениях, мы изучим на следующей лекции.