МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ
Кафедра методов оптимального управления

ДЕЦЕНТРАЛИЗОВАННОЕ ОПТИМАЛЬНОЕ УПРАВЛЕНИЕ

Курсовая работа

Лозовский Иван Иванович
студента 4 курса,
специальность «прикладная математика»

Научный руководитель:
канд. физ.-мат. наук
доцент Н.М. Дмитрук

Минск, 2020

ОГЛАВЛЕНИЕ

ГЛАВА 1

ОСНОВНЫЕ ПОНЯТИЯ И ОБЗОР ЛИТЕРАТУРЫ

В настоящей главе формируются основные понятия, используемые в курсовой работе: приводится классификация задач оптимального управления, разбор их составляющих, определяется объект исследования; даются точные определения программного и позиционного решения; описывается алгоритм работы оптимального регулятора; рассматриваются прямые методы решения задачи оптимального управления, в частности последовательный и параллельный подходы.

1.1 Задачи оптимального управления

Задача оптимального управления формируется из пяти составляющих: временного интервала, математической модели, класса управлений и ограничений на них, ограничений на фазовую траекторию и критерия качества.

1) Временной интервал. По временному интервалу задачи оптимального управления разделяются на непрерывные, рассматриваемые на некотором промежутке времени $T = [t_0,\ t_f]$, и дискретные, где используются дискретные моменты времени $T_h = \{t_0,\ t_0 + h, \ldots, t_f - h\},\ h = \dfrac{t_f - t_0}{N},\ N \in \mathbb{N}$, то есть, например, если $t \in [s,\ s + h[,\ s \in T_h$, то дискретное управление $u(t) = u(s)$. Выделяют задачи с фиксированным и нефиксированном временем окончания динамического процесса, а также задачи на бесконечном интервале.

2) Математическая модель. Динамический процесс обычно моделируется дифференциальными

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

или разностными уравнениями

$$x(k+1) = f(x(k), u(k), k), k = 0, 1, ...,$$

где n-вектор x называется состоянием системы, r-вектор u называется управлением, функция $f : \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{R} \to \mathbb{R}^n$ задана.

3) Класс управлений и ограничения на них. Для непрерывного процесса управления четко указывается класс функций, из которого выбираются управления. Кроме класса доступных управлений задается множество $U \subset \mathbb{R}^r$ — множество допустимых значений управления. Как правило U — компакт в $\mathbb{R}^r$.

Определение 1.1 Кусочно-непрерывная (дискретная, измеримая и т.д.) функция $u(\cdot) = (u(t),\ t \in [t_0,\ t_f])$ называется доступным управлением, если $u(t) \in U,\ t \in [t_0,\ t_f]$.

4) Ограничения на фазовую траекторию. Ограничения на переменные состояния могут накладываться в начальный момент времени t_0:

$$x(t_0) \in X_0;$$

в конечный момент времени t_f, такие ограничения называются терминальными:

$$x(t_f) \in X_f;$$

в изолированные моменты $t_i \in [t_0,\ t_f],\ i = \overline{1, m}$, из промежутка управления — промежуточные фазовые ограничения

$$X(t_i) \in X_i,\ i = 1 \dots m,$$

на всем промежутке управления — фазовые ограничения

$$x(t) \in X(t),\ t \in [t_0,\ t_f],$$

где $X_0,\ X_f,\ X_i,\ i = 1 \dots m,\ X(t),\ t \in [t_0,\ t_f],$ — заданные множества пространства состояний.

Определение 1.2 Доступное управление $u(\cdot) = (u(t),\ t \in [t_0,\ t_f])$ называется допустимым (или, программой), если оно порождает траекторию $x(\cdot)$, удовлетворяющую всем ограничениям задачи.

5) Критерий качества. Множество допустимых управлений, как правило, содержит более одного элемента, поэтому возникает необходимость сравнивать их между собой. Для этого вводится функционал $J(u)$, называемый критерием качества, и выбирается операция минимизации или максимизации этого функционала, результат которой определяет наилучшее(оптимальное) управление. В теории оптимального управления различают четыре типа критериев качества: Майера, Больца, Лагранжа, задачи быстродействия. Все 4 критерия качества эквивалентны между собой.

Для примера выпишем критерий качества типа Майера (териминальный критерий):

$$J(u) = \varphi(x(t_f)).$$

Определение 1.3 Допустимое управление $u^0(\cdot)$ называется оптимальным, если на нем критерий качества достигает экстремального значения.

1.2 Программные и позиционные решения

Объектом исследований в настоящей работе будут непрерывные задачи оптимального управления линейными нестационарными системами с линейным терминальным ограничением и критерием качества:

$$J(u) = c'x(t_f) \to \min, \tag{1.1}$$

$$\dot{x} = A(t)x + B(t)u,\ x(t_0) = x_0,$$

$$x(t_f) \in X_f,$$

$$u(t) \in U,\ t \in T = [t_0, t_f],$$

где $A(t)$ — непрерываная $n \times n$-матричная функция, и $B(t)$ — непрерываная $n \times r$-матричная функция.

Задача (1.1) будет исследоваться в классе дискретных управляющих воздействий

$$u(t) \equiv u(\tau), \quad t \in [\tau,\ \tau + h[, \quad \tau \in T_h = \{t_0,\ t_0 - h, \ldots,\ t_f - h\},$$

где $h = \dfrac{t_f - t_0}{N}$ — период квантавания, $N \in \mathbb{N}$ — заданная мощность множества T_h.

Определение 1.4 Программа $u^0(t),\ t \in T$, называется программным решением задачи (1.1) (оптимальной программой), если на соответствующей ей траектории $x^0(t),\ t \in T$, выполняется равенство $c'x^0(t_f) = \min_u c'x(t_f)$.

Приведем задачу (1.1) к набору задач, зависящих от скаляра $\tau \in T_h =$

$\{t_0,\ t_0 - h, \ldots,\ t_f\}$ и n-вектора z:

$$J(u) = c'x(t_f) \to \min, \tag{1.2}$$

$$\dot{x} = A(t)x + B(t)u,\ x(\tau) = z,$$

$$x(t_f) \in X_f,$$

$$u(t) \in U,\ t \in T(\tau) = [\tau, t_f],$$

Пусть $u^0(t|\tau, z),\ t \in T(\tau),$ — оптимальная программа задачи (1.2) для позиции (τ, z); X_τ — множество состояний z, для которых в момент τ существуют программные решения.

Определение 1.5 Функиция

$$u^0(\tau, z) = u^0(\tau|\tau, z), \quad z \in X_\tau, \quad \tau \in T_h,$$

называется позиционным решением задачи (1.2) (оптимальной обратной связью).

Управление называется программным, если оно регулируется программно, строго, без динамического наблюдения за состоянием объекта и контроля воздействия на него, то есть базируясь только на априорных оценках. В случае позиционного управления управляющие воздействия представляют собой функции от позиции объекта, которые содержат всю доступную на текущий момент информацию. Они также не корректируются в процессе управления.

Программное управление редко применяется на практике, так как со временем, из-за изначальной неточности математической модели и построения обратных связей, а также из-за действия в процессе управления неизвестных возмущений, накапливается общая погрешность вычислений.

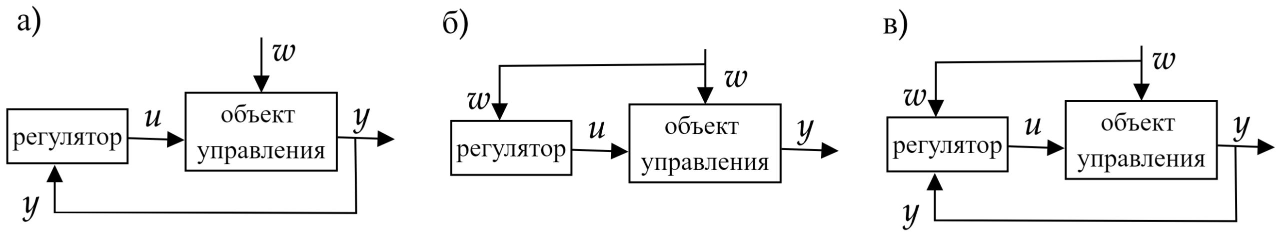

Рис. 1.1: а) обратная связь; б) прямая связь; в) комбинированная связь

Программное и позиционное управления следуют одному из трех принципов управления: по разомкнутому контуру, по замкнутому контуру, в реальном времени. Программные управления исполняются на разомкнутом контуре, а позиционные — на замкнутом и в реальном времени. При создании систем управления по принципу замкнутого контура используются связи 3-х типов: прямые (по входу), обратные (по выходу) и комбинированные (Рис. 1.1). По сути связи — функции, преобразующие наблюдаемые входные и выходные сигналы в управляющие воздействия.

В системах реального времени связи не используются. Нужные для управления их текущие значения вычисляются по ходу каждого процесса управления вычислительными устройствами.

Замкнутые системы управления и системы управления в реальном времени называют автоматическими и автоматизированными, соответственно.

Проблему синтеза оптимальных систем в рамках принципа управления по замкнутому контуру не удается решить из-за проклятия размерности ни с помощью принципа максимума, ни с помощью динамического программирования Беллмана — втрого фундаментального метода теории оптимального управления. Исключение составляет линейно-квадратичная задача Летова-Калмана. В силу этого позиционное решение задачи получается в ви- де простейшей (линейной) обратной связи.

Одним из способов избежания проклятия размерности является переход к синтезу оптимальных систем, следуя современному принципу оптимального управления в реальном времени.

1.3 Управление в реальном времени

Пусть $x^*(\tau)$ — измеренное в конкретном процессе управления состояние объекта управления. Оно отличается от состояния $x(\tau)$ математической модели (1.1) в силу неучтенных в принятой модели возмущений, неточностей математического моделирования, невязки линеаризации нелинейной модели и других факторов.

Введем $\Delta(\tau) \equiv \Delta(\tau,\ x^*(\tau))$ — время отыскания оптимальной обратной связи $u^0(\tau,\ x^*(\tau))$.

Определение 1.6 Функцию $u^*(t),\ t \in T$:

$$u^*(t) \equiv u^0(\tau,\ x^*(\tau)) = u^0(\tau|\tau, x^*(\tau)), \quad t \in [\tau+\Delta(\tau), \tau+h+\Delta(\tau+h)[, \quad \tau \in T_h,$$

назовем реализацией оптимальной обратной связи $u^0(\tau,\ x^*(\tau))$, $z \in X_\tau$, $\tau \in T_h$, в конкретном процессе управления.

Тогда можно сказать, что в момент $\tau \in T_h$ определяется состояние объекта, а в момент $\tau + \Delta(\tau)$ будет найдено оптимальное для момента τ управление, которое подается на вход объекта управления.

Определение 1.7 Если в каждый момент времени $\tau \in T_h$ вычисление $u^*(\tau)$ производится за время $\Delta(\tau) < h$, то описанная выше схема управления объектом называется управлением в реальном времени.

Определение 1.8 Оптимальным регулятором, реализующим оптимальную обратную связь, называется устройство, способное вычислять $u^*(\tau),\ \tau \in T_h$, за время $\Delta(\tau) < h$.

Далее, в главе 2, будет приведен конкретный алгоритм работы оптимального регулятора.

1.4 Численные методы решения задач оптимального управления и программные средства

Различают несколько подходов к решению непрерывных задач оптимального управления. Отметим динамическое программирование [?], непрямые методы, основанные на применении принципа максимума [?] и прямые методы решения. В настоящей работе будут применяться последние.

Прямые методы сводят непрерывную динамическую систему к системе с дискретным временем, после чего применяются численные методы нелинейной оптимизации (или линейного, квадратичного программирования).

Рассмотрим прямые методы на примере методов Single Shooting и Multiple Shooting, которые в свою очередь реализуют последовательный и параллельный подходы решения задач оптимального управления, соответственно.

Все Shooting методы содержат в себе модули для решения ОДУ, что позволяет исключить динамическую систему в непрерывном времени. Это осуществляется через замену функции управления $u(t)$ полиномами, кусочно-постоянной функцией или сплайном.

Обозначим конечное множество параметров управления вектором q, а итоговую функцию управления как $u(t;\ q)$.

Наиболее распространенная форма управления — кусочно-постоянные управления, для которых выбирается фиксированная сетка $0 = t_0 < t_1 < ... < t_N = t_f$ и N параметров $q_i \in \mathbb{R}^{n_u}$, $i = 0, ..., N-1$. При этом полагается, что $u(t;\ q) \equiv q_i, \quad t \in [t_i,\ t_{i+1}]$. Если сетка равномерная, то рассматриваемое управление — дискретное.

Таким образом размерность вектора $q = (q_0,\ \dots\ , q_{N-1})$ — $N \times n_u$.

В методе Single Shooting, являющемся последовательным подходом, $x(t)$, $[0, t_f]$, находится с помощью численного метода решения ОДУ, в котором начальное условием задается через x_0 и используются значения $u(t;\ q)$. Итоговую траекторию обозначим как $x(t;\ q)$, $t \in [0, t_f]$.

Тогда исходная задача примет вид:

$$c' x(t_N;\ q) \to \min_q,$$

$$x(t_0, q) = x_0,$$

$$x(t_N;\ q) \in X_f,$$

$$q_i \in U,\ i = 0, 1, \dots, N-1.$$

Полученная задача — задача математического программирования. Если X_f, U — многогранники, то это задача линейного программирования.

В методе Multiple Shooting, являющемся параллельным подходом, аналогично Single Shooting методу управление дискретизируется на сетке:

$$u(t;\ q) \equiv q_i, \quad t \in [t_i,\ t_{i+1}].$$

Однако в дальнейшем ОДУ решается отдельно для каждого интервала $[t_i,\ t_{i+1}]$, с заданными на них искусственными начальными значениями состояния s_i:

$$\dot{x}_i(t;\ s_i,\ q_i) = f(x_i(t;\ s_i,\ q_i),\ q_i),$$

$$t \in [t_i,\ t_{i+1}],$$

$$s_0 = x_0,$$

$$s_{i+1} = x_i(t_{i+1};\ s_i,\ q_i).$$

В итоге получм оптимизационную задачу вида:

$$c' s_N \to \min_{q,s},$$

$$s_0 - x_0 = 0,$$

$$x_i(t_{i+1};\ s_i,\ q_i) - s_{i+1} = 0,$$

$$q_i \in U,\ i = 0, 1, \dots, N-1.$$

Таким образом определен, объект исследования настоящей работы — это линейная задача терминального управления. На примере построенной задачи введены понятия позиционного и программного решения, описан принцип работы оптимального регулятора. На примере данной задачи показано, как реализуются последовательный и параллельный подходы решения непрерывных задач оптимального управления и описаны методы Single Shooting и Multiple Shooting.

ГЛАВА 2

ЦЕНТРАЛИЗОВАННОЕ ОПТИМАЛЬНОЕ УПРАВЛЕНИЕ

(Врезка) ...

2.1 Задача оптимального управления группой динамических систем

Определим T и T_h как в первой главе: $T = [t_0,\ t_f]$, $T_h = \{t_0,\ t_0 - h, \ldots,\ t_f - h\}$, $h = \dfrac{t_f - t_0}{N}$, $N \in \mathbb{N}$. Пусть $I = \{1,\ 2,\ ...,\ q\}$, $I_i = I \backslash i$; $A_{ij}(t) \in R^{n_i \times n_j}, B_{ij}(t) \in R^{n_i \times r_j}$, $t \in T$, $i, j \in I$, — кусочно-непрерывные матричные функции; $A_i(t) = A_{ii}(t)$, $B_i(t) = B_{ii}(t)$, $t \in T$, $i \in I$. $u_{i*}, u_i^* \in R^{r_i}$ - заданные векторы

Составим задачу для группы q взаимосвязанных объектов управления.

Будем считать, что на промежутке T модель i-го ($i \in I$) имеет вид:

$$\dot{x}_i = A_i(t)x_i + \sum_{j \in I_i} A_{ij}(t)x_j + B_i(t)u_i + \sum_{j \in I_i} B_{ij}(t)u_j, \quad x_i(t_0) = x_{i0},$$

где $x_i = x_i(t) \in R^{n_i}$ — состояние i-ой математической модели; $u_i = u_i(t) \in U_i = \{u \in R^{r_i} : u_{i*} \leq u \leq u_i^*\}$ — дискретное управляющие воздействие i-ой математической модели с периодом квантования h. Функции $A_{ij}(t), t \in T, i \in I_j$ служат для описания влияния на i-ую модель остальных моделей, а функции $B_{ij}(t), t \in T, i \in I_j$ — для описания влияния на на i-ую модель управляющих воздействий остальных моделей.

Определим терминальные ограничения:

$$x_i(t_f) \in X_{fi}, i \in I; \quad X_f = X_{f1} \times X_{f2} \times ... \times X_{fq}.$$

Целью управления является минимизация терминального критерия качества:

$$\sum_{i \in I} c_i' x_i(t_f) \to \min_{u=(u_1', \ldots, u_q')'},$$

где $c_i \in R^{n_i}$ — заданные векторы.

Группой динамических объектов можно управлять централизованно и децентрализованно. В первом случае имеется общий центр управления, которой на каждом промежутке времени $[\tau, \tau+h[, \tau \in T_h$ по точной информации о состоянии $x^*(\tau) = (x_i^*(\tau), i \in I)$ группы вырабатывает управляющие воздействия $u^*(t) = (u_i^*(t), i \in I), t \in [\tau, \tau+h[, \tau \in T_h$, которое используется до тех пор, пока не будет измерено и обработано следующее состояние $x^*(\tau+h)$. Во втором случае для каждого i-го объекта по точному состоянию $x_i^*(\tau), \tau \in T_h$ и состояниям $x_j^*(\tau - h), \tau \in T_h, j \in I_i$ на локальном уровне строится управляющее воздействие $u_i^*(t), t \in [\tau, \tau+h[, \tau \in T_h$, которое используется до тех пор, пока не будет измерено и обработано следующее состояние $x_i^*(\tau+h)$.

При централизованном управлении динамическая модель задачи рассматривается как одна большая система:

$$\dot{x} = A(t)x + B(t)u, \quad x(0) = x_0,$$

где $x = (x_1', ..., x_q')', u = (u_1', ..., u_q')'$; $A,\ B$ — соответствующие блочные матрицы.

Система имеет единственный регулятор, который в режиме реального времени на основе измеренного точного текущего состояния $x^*(\tau)$ вырабатывает управляющий сигнал $u^*(\tau) \in R^{r_1+r_2+...+r_q}$. Для этого оптимальный регулятор для каждого момента $\tau \in T_h$ решает задачу Task(τ):

$$c'x(t_f) \to \min_u,$$

$$\dot{x} = A(t)x + B(t)u, \quad t \in [\tau, t_f],$$

$$x(\tau) = x^*(\tau)$$

$$x(t_f) \in X_f, \quad u \in U,$$

где $c' = (c_1', ..., c_q')'$; $U = U_1 \times U_2 \times ... \times U_q$.

После сведения исходной задачи к задаче линейного программирования, получим:

$$c'x(t_N, v) \to \min_v, x(t_0, v) = x_0, x(t_f) \in X_f, v \in U,$$

где $v = (v_0, ..., v_{N-1})$ — характеризующий u вектор параметров, $x(t, v)$ — зависящая от v итоговая траектория. (я не понял, что надо сделать и еще кое-что...)

2.2 Построение централизованной обратной связи в реальном времени

.....

Не забываем делать выводы.