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

ПОСТРОЕНИЕ НЕЙРОРЕГУЛЯТОРА ДЛЯ ЗАДАЧИ УПРАВЛЕНИЯ КВАДРОКОПТЕРОМ

Курсовая работа

Синицын Игорь Олегович
студента 3 курса,
специальность «прикладная математика»

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

Минск, 2020

ОГЛАВЛЕНИЕ

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Курсовая работа, 22 с., 4 рис., 11 источников

КЛЮЧЕВЫЕ СЛОВА ОПТИМАЛЬНОЕ УПРАВЛЕНИЕ, УПРАВЛЕНИЕ С ПРОГНОЗИРУЮЩЕЙ МОДЕЛЬЮ, MPC-РЕГУЛЯТОР, КВАДРОКОПТЕР

Объектом исследования курсовой работы являются задачи стабилизации нелинейных динамических систем, на управляющие воздействия и траектории которых наложены ограничения, управление по прогнозирующей модели.

Целью работы является построение MPC-регулятора и симуляция его работы в задаче управления квадрокоптером.

Работа состоит из трех глав. В первой главе определяются основные понятия и результаты в области управления с прогнозирующей моделью. Во второй главе рассматриваются методы построение MPC-регулятора, связанные с методами машинного обучения. В третьей главе проиллюстрированы результаты построение MPC-регулятора в задаче управления квадрокоптером.

Областями применения MPC-регуляторов могут быть, как в данной курсовой работе, системы управления беспилотных летательных средств, транспортные системы, робототехника, химическая промышленность и др.

ВВЕДЕНИЕ

Model Predictive Control (MPC) — удобный механизм решения задач оптимального управления в режиме реального времени, позволяющий эффективно решать задачи с ограничениями на траекторию или управление, что позволяет сократить потенциально потраченные на управление линейной или нелинейной системой ресурсы. Однако в задачах, которые требует активного реагирования на быструю смену состояния, MPC может не успевать решать поставленную ему задачу оптимизации в отведенные сроки, до следующего изменения состояния системы.

В главе 1 будут рассмотрены основные принципы управления по прогнозирующей модели, базовый алгоритм MPC, задачи оптимального управления и описаны численные методы, позволяющие решить или сократить решение задач оптимального управления. В главе 2 ознакомимся с одним из способов машинного обучения — обучением с подкреплением, рассмотрим методы построения MPC-регулятора, где может быть использовано обучение с подкреплением. В третьей главе будет рассмотрена математическая модель квадрокоптера, его нелинейная динамика, будут показаны этапы построения MPC-регулятора, проведена симуляция полученной системы, и проиллюстрированы результаты.

ГЛАВА 1

ОСНОВНЫЕ ПОНЯТИЯ И ОБЗОР ЛИТЕРАТУРЫ

Управление по прогнозирующей модели (Model Predictive Control) — продвинутый способ управления линейными и нелинейными системами, основанный на решении задач оптимального управления с конечным горизонтом, удовлетворяя ограничениям, в режиме реального времени.

В этой главе описываются основные принципы MPC, базовый алгоритм, рассматриваются задачи оптимального управления и описываются некоторые численные методы решения ЗОУ.

1.1 Теория управления по прогнозирующей модели

1.1.1 Прогнозирующая модель

Рассмотрим нелинейную дискретную прогнозирующую модель вида:

$$x^+ = f(x, u), \tag{1.1}$$

где функция $f : X \times U \longrightarrow X$ ставит в соответствие паре $x \in \mathbb{X}$ и $u \in \mathbb{U}$ некоторое состояние $x^+ \in \mathbb{X}$, которое является состоянием системы в последующий момент времени.

Траектория системы (1.1) может быть получена следующим образом: с заранее заданным значением $x_0 \in \mathbb{X}$, а также с управлением $u(\cdot) \in \mathbb{U}^k,\ k \in \mathbb{N}_\infty$, можно определить значение траектории $x_u(k)$ итерационно через

$$x(0) = x_0,\ x_u(k+1) = f(x_u(k), u(k)), \tag{1.2}$$

для всех $k \in \mathbb{N}_0$, если $k = \infty$, и $k = 0, 1, ..., k-1$ в другом случае.

1.1.2 Ограничения

Одна из основных частей MPC — ограничения. Они могут быть установлены как на управление, так и на состояние системы. Введем непустое множество $X \subseteq \mathbb{X}$ и для каждого $x \in \mathbb{X}$ введем непустое множество ограничений $\mathbb{U}(x) \subseteq U$. Множество ограничений $\mathbb{U}(x)$ может и не зависеть от

x. За введением ограничений стоит наше желание указать, что траектория лежит во множестве $\mathbb{X}$, а соответствующее траектории управление лежит во множестве $\mathbb{U}(x)$.

1.1.3 Критерий качества

Пусть $x_* \in X$ — точное решение системы (номер). Тогда функция стоимости, которая используется в нашем процессе оптимизации, в критерии качества, должна возвращать значение, основанное на растоянии от любого $x \in X$ до x_*. В дополнение к уже сказанному, желательно, чтобы функция стоимости также включала в себя оценку расстояний и по управлению $u \in U$. По вычислительным причинам намного удобнее, чтобы управление также оценивалось функцией стоимости. Также оценка управления может быть полезна с точки зрения моделирования, т.к. мы не хотим использовать управление, затратное по ресурсам или энергии. Благодаря этим причинам, мы выбираем функцию стоимости вида $l : X \times U \longrightarrow \mathbb{R}_0^+$.

Из всего вышесказанного, можно сделать вывод, что функция стоимости обладает следующими свойствами:

$$l(x_*, u_*) = 0, \tag{1.3}$$

$$l(x, u) > 0,\ x \in X,\ u \in U,\ x \neq x_*. \tag{1.4}$$

1.1.4 Базовая формулировка задачи оптимального программирования с конечным горизонтом

Базовая формулировка задачи оптимального программирования с конечным горизонтом имеет вид: необходимо минимизировать критерий качества на траектории нелинейной дискретной системы с ограничениями на управление

$$J_N(x_0, u(\cdot)) = \Sigma_{k=0}^{N-1} l(x_u(k, x_0), u(k)) \longrightarrow min, \tag{1.5}$$

$$x_u(k+1, x_0) = f(x_u(k, x_0), u(k)),\ x_u(0, x_0) = x_0, \tag{1.6}$$

$$u(\cdot) \in \mathbb{U}^N(x_0). \tag{1.7}$$

1.1.5 Базовый алгоритм MPC

Идея схемы MPC выглядит так: в каждый момент квантования n оптимизируем предсказанное поведение системы и используем первый элемент результирующей последовательности управлений в качестве значения оценки в следующий момент квантования. Таким образом, базовый алгоритм MPC будет иметь следующий вид:

в каждый момент квантования $t_n,\ n = 0, 1, 2, ...$

1. вычисляем состояние $x(n) \in X$ системы (номер),
2. принимаем $x_0 = x(n)$ и решаем задачу оптимального управления, сформулированную в предыдущем пункте 1.1.4 (номер - номер), после чего обозначим полученную последовательность оптимальных управлений через $u^*(\cdot) \in \mathbb{U}^N(x_0)$,
3. определим ***оценочное значение*** $\mu_N(x(n)) = u^*(0) \in U$ и используем это значение управления в следующем периоде квантования.

1.2 Задачи оптимального управления

1.2.1 Классификация задач оптимального управления

Задачи оптимального управления классифицируются по следующим составляющим:

- Промежуток времени и управления.

  Промежуток времени в задачах оптимального управления может быть как непрерывным, т.е. $t \in \mathbb{R}$, так и дискретным, а именно $k \in \mathbb{Z}_+$. В то же время, промежуток управления может быть конечным, что значит $t \in [t_0, t_f],\ t_0 < t_f$, и бесконечным, где $t \in [t_0, +\infty)$. Время окончания процесса в свою очередь также может быть подразделено на фиксированное, т.е t_f заданно, и не фиксированное, где t_f уже заданно не будет.

- Математическая модель объекта управления.

  Будем рассматривать объекты управления, которые можно описать обыкновенными дифференциальными уравнениями:

  $$\dot{x} = f(x, u, t), \tag{1.8}$$

  где $x = x(t) \in \mathbb{R}^n$ — состояние системы, $u = u(t) \in \mathbb{R}^r$ — управление.

- Классы управлений и ограничение на управление.

  Классы управлений могут подразделяться на кусочно-непрерывные, измеримые, дискретные, релейные, инерционные и др. функции.

Ограничение на управление:

$u(t) \in U(t),\ t \in [t_0, t_f]$, где $U(t) \subseteq \mathbb{R}^n$.

- Ограничения на траекторию.

  В общем виде ограничения на траекторию имеют вид:

$$x \in \mathbb{X}(t),\ t \in [t_0, t_f]. \tag{1.9}$$

  Ограничения на траекторию могут накладываться:

  - на правом конце траектории (терминальные ограничения), т.е. $x(t_f) \in \mathbb{X}_f$, откуда могут вытекать задачи с закрепленным, свободным и подвижным правым концом траектории;
  - на левом конце траектории, т.е. $x(t_0) \in \mathbb{X}_0$, откуда также могут вытекать задачи с закрепленным, свободным и подвижным концом траектории;
  - в промежуточные моменты времени, т.е. $x(t_i) \in \mathbb{X}_i,\ t_i \in [t_0, t_f],\ i = 1, ..., l$, при этом $t_0 < t_1 < ... < t_l < t_f$.

  Существуют и смешанные ограничения на траекторию.

- Критерий качества.

  - Терминальный критерий (критерий качества типа Майера) $J(u) = \varphi(x(t_f)),\ \varphi : \mathbb{R}^n \longrightarrow \mathbb{R}$;
  - Интегральный критерий (критерий качества типа Лагранжа) $J(u) = \int_{t_0}^{t_f} f_0(x(t), u(t), t)dt,\ f_0 : \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{R} \longrightarrow \mathbb{R}$;
  - Критерий качества типа Больца $J(u) = \varphi(x(t_f)) + \int_{t_0}^{t_f} f_0(x(t), u(t), t)dt$;
  - Критерий быстродействия $J(u) = t_f - t_0 \longrightarrow min$.

1.2.2 Принцип максимума для простейшей задачи оптимального управления

Рассмотрим простейшую задачу оптимального управления на промежутке времени $[t_0, t_f]$ в классе кусочно-непрерывных управлений:

$$J(u) = \varphi(x(t_f)) \longrightarrow min, \tag{1.10}$$
$$\dot{x} = f(x, u, t),\ x(t_0) = x_0, \tag{1.11}$$
$$u(t) \in \mathbb{U},\ t \in [t_0, t_f]. \tag{1.12}$$

В задаче (номер откуда до куда) необходимо минимизировать критерий типа Майера (номер) на траектории нелинейной системы (номер) с ограничениями на управление (номер).

Принцип максимума.

Пусть $u^0(t),\ t \in [t_0, t_f]$ — оптимальное управление, $x^0(t),\ t \in [t_0, t_f]$ — оптимальная траектория, $\psi^0(t),\ t \in [t_0, t_f]$ — сопряженная траектория — решение сопряженного уравнения

$$\psi' = -\frac{\partial H(x^0(t), \psi, u^0(t), t)}{\partial x}, \tag{1.13}$$
$$\psi = -\frac{\partial \varphi(x^0(t_f))}{\partial x}. \tag{1.14}$$

Тогда необходимо выполняется условие максимума гамильтониана:

$$H(x^0(t), \psi^0(t), u^0(t), t) = \max_{u \in \mathbb{U}} H(x^0(t), \psi^0(t), u, t),\ t \in [t_0, t_f]. \tag{1.15}$$

1.2.3 Алгоритм решения задач оптимального управления с помощью принципа максимума

Алгоритм решения задачи оптимального управления (номер-номер) из предыдущего пункта состоит из:

- составления гамильтониана

  $$H(x, u, \psi, t) = \psi' f(x, u, t), \tag{1.16}$$

  где $\psi = \psi(t) \in \mathbb{R}^n$;

- записи сопряженной системы и условия трансверсальности

  $$\dot{\psi} = -\frac{\partial H(x, u, \psi, t)}{\partial x}, \tag{1.17}$$
  $$\psi(t_f) = -\frac{\partial \varphi(x(t_f))}{\partial x}; \tag{1.18}$$

- условия максимума гамильтониана

$$H(x^0(t), \psi^0(t), u^0(t), t) = \max_{u \in \mathbb{U}} H(x^0(t), \psi^0(t), u(t), t),\ t \in [t_0, t_f]; \quad (1.19)$$

- рассмотрения задачи максимума гамильтониана

$$H(x, \psi, u, t) \longrightarrow max, \quad (1.20)$$

  где x, ψ, t предполагаются фиксированными.

- результата решения предыдущего пункта

$$u(x, \psi, t) = arg \max_{u \in \mathbb{U}} H(x, \psi, u, t); \quad (1.21)$$

- подстановки результата в прямую и сопряженную систему

$$\dot{x} = f(x, u(x, \psi, t), t), \quad (1.22)$$

$$\dot{\psi} = -\frac{\partial H(x, \psi, u(x, \psi, t), t)}{\partial x}, \quad (1.23)$$

$$x(t_0) = x_0,\ \psi(t_f) = -\frac{\partial \varphi(x(t_f))}{\partial x}; \quad (1.24)$$

- решения краевой задачи принципа максимума (номер - номер)

$$x(t), \psi(t),\ t \in [t_0, t_f], \quad (1.25)$$

$$u(t) = u(x(t), \psi(t), t),\ t \in [t_0, t_f]. \quad (1.26)$$

1.3 Численные методы решения задач оптимального управления и программные средства

Рассмотрим некоторые численные методы решения задач оптимального управления:

- Численное решение задач оптимального управления с переключением методом пристрелки.

  Система дифференциальных уравнений состояния объекта вместе с сопряженной системой, краевыми условиями и принципом максимума Понтрягина составляют П-систему. Интегрирование П-системы осуществляется методом Рунге-Кутта четвертого порядка. Для решения уравнений невязок и подбора уточненных значений параметров пристрелки используется метод Ньютона.

- Редукция к задаче нелинейного программирования.

  Для использования этого метода необходимо перейти от непрерывной системы к дискретной. Отрезок $[0, T]$ разбивается на $(k-1)$ участок, где система $\dot{x} = f(x, t)$, будучи уже дискретной и имеющей вид $x(t_i) = \eta(x(t_i), \Delta u(t_i), t_i)$, интегрируется по схеме, относящейся к семейству методов Рунге-Кутта. Ограничения заменяются их дискретными аналогами. В результате задача оптимального управления сводится к задаче нелинейного программирования.

- Метод последовательных приближений.

  В качестве начального приближения $u_0(t)$ выбирается некоторое допустимое управление. Обычно при его выборе используются физические соображения. Метод состоит из последовательных итераций. На i-ой итерации находится решение задачи Коши $x_i(t)$. Затем интегрированием в обратном времени находится решение $p_i(t)$ сопряженной системы. После чего, управление $u_{i+1}(t)$ определяется из условия максимума, и осуществляется переход к следующей итерации. Процесс продолжается до тех пор, пока на двух последующих итерациях не будет достигнута требуемая точность.

1.4 Выводы

Идея MPC: оптимизация будущего поведения системы в каждый момент времени, нахождение оптимального управления и его использования в качестве значения обратной связи для следующего момента времени. Из-за простоты идеи и реализации, MPC получил большое распространение в различного рода промышленных приложениях.

ГЛАВА 2

МЕТОДЫ ПОСТРОЕНИЯ MPC-РЕГУЛЯТОРОВ НА ОСНОВЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

В настоящей главе ознакомимся с одним из методов машинного обучения — обучением с подкреплением, опишем базовые принципы его работы. Рассмотрим методы построения MPC-регуляторов на основе методов машинного обучения.

2.1 Обучение с подкреплением

Рассмотрим один из способов машинного обучения — обучение с подкреплением (reinforcement learning). Суть этого метода обучения заключается в том, что наша система (агент) взаимодействует с некоторой средой, которая, в свою очередь, реагирует на действия агента, посылая сигналы подкрепления. Обучение с подкреплением является частным случаем обучения с учителем, однако главной особенностью RL является то, что учитель — это среда или ее модель.

Базовое обучение с подкреплением моделируется как процесс принятия решения Маркова:

- набор состояний среды и агента S
- набор действий агента A
- $P_a(s,s') = Pr(s_{t+1} = s'|s_t = s, a_t = a)$ — вероятность перехода из состояния s в состояние s' под действием агента a
- $R_a(s,s')$ — награда за переход из состояния s в состояние s' под действием агента a.

Агент в каждый дискретный момент времени взаимодействует со средой. У агента имеется либо полный, либо частичный доступ к просмотру множества состояний. После действия, которое может быть произведено случайным образом, агент получает награду за выбранное им действие. Цель обучения с подкреплением связана с получением наибольшего количества наград. Выбор действия агентом может основываться не только на награде в данный момент времени, но и на награде, которую он получит после выбора этого действия. Это говорит о том, что обучение с подкреплением хорошо вписывается в задачи с долгосрочной перспективой.

2.2 Методы построения MPC-регуляторов на основе методов машинного обучения

Существует некоторое количество принципов использования методов машинного обучения в MPC:

- Явный MPC
- Аппроксимация закона управления
- Использование обучаемой модели для аппроксимации динамики прогнозирующей модели
- Итерационный подход для построения терминального региона и функции из предыдущих итераций.

Рассмотрим явный MPC. Для линейных систем задача оптимизации может быть решена до начала процедуры управления. В результате решения получаем явный закон управления $u(x)$. Необходимо найти $x(t)$ в момент времени t в задаче вида

$$V(x) = \min_{u(\cdot|t)} \sum_{k=t}^{t+N-1} L(x(k|t), u(k|t)) + F(x(t+N|t)), \tag{2.1}$$
$$x(k+1|t) = Ax(k|t) + Bu(k|t),\ t \le k \le t+N-1, \tag{2.2}$$
$$x(t|t) = x(t),\ t \le k \le t+N-1, \tag{2.3}$$
$$C_x x(k|t) \le d_x,\ t \le k \le t+N-1, \tag{2.4}$$
$$C_u u(k|t) \le d_u,\ t \le k \le t+N-1, \tag{2.5}$$
$$C^f x(t+N|t) \le d_f, \tag{2.6}$$

где $L(x(k|t), u(k|t))$ и $F(x(k+N|t))$ квадратичные функция стоимости и терминальная функция соответственно, имеющие вид

$$L(x(k|t), u(k|t)) = x(t)^T Q x(t) + u(t)^T R u(t),\ Q, R > 0, \tag{2.7}$$
$$F(x(k+N|t)) = x(t)^T P x(t). \tag{2.8}$$

Перепишем задачу (2.1) - (2.6) в матричном виде:

$$\min_U \frac{1}{2}U^T HU + x^T FU + \frac{1}{2}x^T Yx, \tag{2.9}$$

$$GU \leq W + Ex(t) \tag{2.10}$$

$$
\begin{aligned}
&Y = 2(Q + \Omega^T \tilde{Q}\Omega),\ H = 2(\Gamma^T \tilde{Q}\Gamma + \tilde{R}),\ F = 2\Omega^T \tilde{Q}\Gamma,\\
&\tilde{Q} = diag(Q, ..., Q, P) \in \mathbb{R}^{n\times(N+1)},\ \tilde{R} = diag(R, ..., R) \in \mathbb{R}^{m\times N},\\
&\Omega = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix},\ \Gamma = \begin{bmatrix} B & 0 & & \dots & 0 \\ AB & B & 0 & \dots & 0 \\ \dots & & & & \\ A^{N-1}B & A^{N-2}B & & \dots & 0 \end{bmatrix},\\
&G = diag(C_x, \dots, C_x)\Gamma,\ W = [d_x, \dots, d_x]^T,\ E = diag(C_x, \dots, C_x)\Omega.
\end{aligned}
$$

Сделав замену переменных вида
$z = U + H^{-1}F^T x$, уравнения (2.9) - (2.10) примут вид

$$\min_z \frac{1}{2}z^T Hz + \frac{1}{2}x^T \tilde{Y}x, \tag{2.11}$$

$$Gz \leq W + Sx, \tag{2.12}$$

$$\tilde{Y} = Y - FH^{-1}F^T, \tag{2.13}$$

$$S = E + GH^{-1}F^T. \tag{2.14}$$

Далее эта задача решается с помощью теории выпуклой оптимизации через условия Каруша-Куна-Такера.

Явный MPC решает задачу для всех состояний, разбивая все пространство состояний на области, в каждой из которых находится своя явная функция управления.

Алгоритм нахождения явных функций управления:

1. Взять любой $x_0 \in \mathbb{X}$

2. Решить задачу (2.11) - (2.12) с начальным условием $x = x_0$

3. Определить активные ограничения для оптимизационной задачи (2.11) - (2.12)

4. Вычислить критическую область по активным ограничеям и вычислить функцию управления для этой области

5. Перейти к новому x_0

2.3 Выводы

Каждый из методов имеет свое собственное применение в определенной ситуации. Рассмотренный метод имеет одним большим недостатком — потенциально большим количеством областей, что может плохо сказаться на производительности метода. Однако для линейных задач с небольшим количеством областей явный MPC будет подходить.

ГЛАВА 3

ПОСТРОЕНИЕ MPC-РЕГУЛЯТОРА В ЗАДАЧЕ УПРАВЛЕНИЯ КВАДРОКОПТЕРОМ

В текущей главе рассмотрим математическую модель квадрокоптера, этапы, необходимые для формулирования решаемой MPC-регулятором задачи оптимизации, построение MPC-регулятора с помощью Model Predictive Control Toolbox, рассмотрим структуру системы, ее настройку, и проведем симуляцию полученной модели в Simulink.

3.1 Математическая модель квадрокоптера

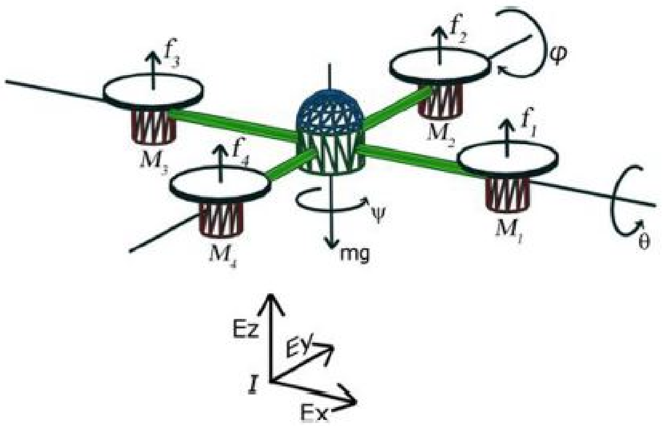

Рис. 3.1: Модель квадрокоптера

Квадрокоптер может свободно двигаться во всех трех направлениях во время своего полета. Это значит, что нам необходимо ввести шесть степеней свободы, чтобы полностью описать динамику квадрокоптера. Обозначим с помощью x, y, z позицию квадрокоптера в пространстве, где x и y отвечают за горизонтальную плоскость, а z — за вертикальную. С помощью φ, θ, ψ обозначим углы вращения квадрокоптера относительно своих осей: ψ отвечает за угол поворота (вращение относительно оси z), φ отвечает за крен (вращение относительно оси y), θ отвечает за угол наклона квадрокоптера (вращение относительно оси x).

Каждый из моторов M_1, M_2, M_3, M_4 генерирует силы тяги f_1, f_2, f_3, f_4 и крутящие моменты $\tau_1, \tau_2, \tau_3, \tau_4$ соответственно. Силы и крутящие моменты могут быть изменены путем повышения или понижения напряжения на моторы. В итоге, создаются три крутящих момента $\tau_\varphi, \tau_\theta, \tau_\psi$:

$$\tau_\theta = (f_2 - f_4)l, \tag{3.1}$$

$$\tau_\varphi = (f_1 - f_3)l, \tag{3.2}$$

$$\tau_\psi = \sum_{i=1}^{4} \tau_i, \tag{3.3}$$

где l — это расстояние от каждого из моторов до центра квадрокоптера. А результирующая сила $u = f_1 + f_2 + f_3 + f_4$ позволяет свободно изменять позицию и ориентацию квадрокоптера в пространстве.

Пусть $\tau = [\tau_\theta \;\; \tau_\varphi \;\; \tau_\psi]'$ будет общим вектором крутящих моментов, $\eta = [\theta \; \varphi \; \psi]'$ будет вектором углового смещения, а J — матрица инерции. Тогда вращательная динамика квадрокоптера может быть описана следующим образом:

$$\tau = J\ddot{\eta} + J\dot{\eta} - \frac{1}{2}\frac{\partial}{\partial \eta}(\dot{\eta}' J \dot{\eta}), \tag{3.4}$$

что может быть переписано как

$$\ddot{\eta} = \tilde{\tau}, \tag{3.5}$$

где $\tilde{\tau} = [\tilde{\tau}_\theta \; \tilde{\tau}_\varphi \; \tilde{\tau}_\psi]'$ новые входные данные. С помощью вращательных преобразований между системой I (см. Рис. 3.1) и центром масс квадрокоптера, получим систему уравнений, описывающих его динамику:

$$m\ddot{x} = -u \sin\theta - \beta\dot{x}, \tag{3.6}$$

$$m\ddot{y} = u \cos\theta \sin\varphi - \beta\dot{y}, \tag{3.7}$$

$$m\ddot{z} = u \cos\theta \cos\varphi - \beta\dot{z}, \tag{3.8}$$

$$\ddot{\theta} = \tilde{\tau}_\theta, \tag{3.9}$$

$$\ddot{\varphi} = \tilde{\tau}_\varphi, \tag{3.10}$$

$$\ddot{\psi} = \tilde{\tau}_\psi, \tag{3.11}$$

где β является фактором демпфирования, принимая во внимание эффекты трения. Нелинейная динамическая модель имеет двеннадцать входных данных (шесть позиций и шесть скоростей) и четыре выходных данных (результирующая сила и три крутящих момента), связанных соотношениями (3.6) - (3.11).

3.2 Построение линейного MPC

3.2.1 Формулирование и построение MPC-регулятора

Для того, чтобы построить линейный MPC-регулятор для квадрокоптера, необходимо линеаризовать нелинейную динамическую систему (3.6) - (3.11) в положении равновесия. Результирующее линейное непрерывное пространство состояний нужно перевести в дискретное с периодом квантования T_s

$$\xi(k+1) = A\xi(k) + Bu(k), \tag{3.12}$$
$$y(k) = C\xi(k) + Du(k), \tag{3.13}$$

где $\xi(k) = [x\ \dot{x}\ y\ \dot{y}\ z\ \dot{z}\ \varphi\ \dot{\varphi}\ \theta\ \dot{\theta}\ \psi\ \dot{\psi}]' \in \mathbb{R}^{12}$ — вектор состояния системы, $y(k) \in \mathbb{R}^{12}$ — вектор выходных данных, матрицы A, B, C, D — это матрицы подходящих размеров, полученных в процессе линеаризации системы (3.6) - (3.11).

Для формулирования линейного MPC используется Model Predictive Control Toolbox для Matlab, где управление получено с помощью решения оптимизационной задачи

$$\min_{\Delta u} \sum_{i=0}^{N_L-1} (\sum_{j=1}^{n_y} |\omega_j^y[y_j(k+i+1|k) - r_j(k+i+1)]|^2 + \sum_{j=1}^{n_u} |\omega_j^{\Delta u} \Delta u_j(k+i|k)|^2 + \rho_\epsilon \epsilon^2), \tag{3.14}$$

$$u_j^{min} \leq u_j(k+i|k) \leq u_j^{max},\ j = 1, ..., n_u, \tag{3.15}$$
$$y_j^{min} - \epsilon V_j^{y,min} \leq y_j(k+i+1|k) \leq y_j^{max} + \epsilon V_j^{y,max},\ j = 1, ..., n_y, \tag{3.16}$$
$$\Delta u(k+h|k) = 0,\ h = N_{Lu}, ..., N_L,\ \epsilon \geq 0, \tag{3.17}$$

для всех $i = 0, ..., N_L - 1$ в отношении последовательности входных приращений $\{\Delta u(k|k), ..., \Delta u(N_{Lu} - 1 + k|k)\}$ и слабой переменной ϵ. Индекс "$(\cdot)_j$" обозначает j-ую компоненту вектора, "$(k+i|k)$" обозначает предсказанное значение в момент времени $k+i$, основываясь на текущем моменте времени k, $r(k)$ — значение эталонных выходных данных (output reference), $V^{y,min}, V^{y,max}$ — неотрицательные векторы-константы, необходимые для релаксации выходных ограничений, $n_y = 12$ количество выходных значений, $n_u = 4$ количество входных значений. Линейный MPC-регулятор устанавливает значение $u(k) = u(k-1) + \Delta u^*(k|k)$, где $\Delta u^*(k|k)$ является первым элементом последовательности.

3.2.2 Настройка MPC и проверка результатов

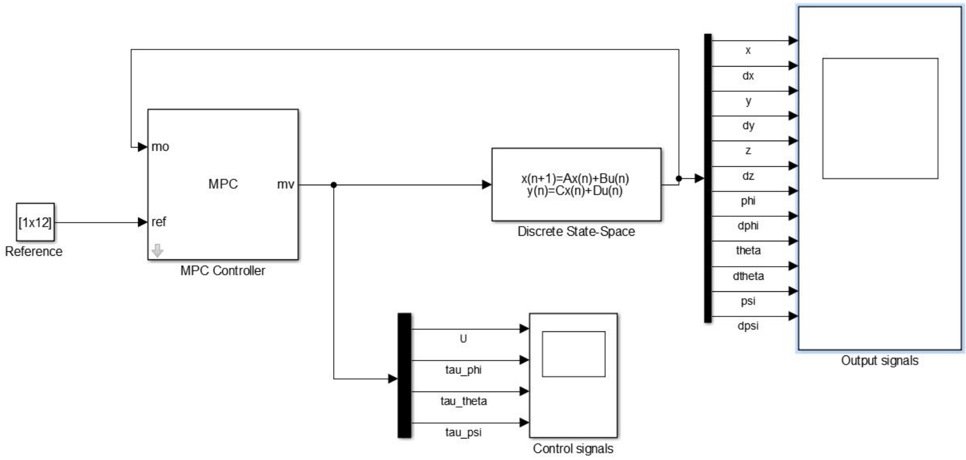

Рис. 3.2: Структура системы

Линейный MPC-регулятор настроен следующим образом. Устанавливаем $u_j^{min} = -6N_m$, $u_j^{max} = 6N_m$, $j = 1, 2, 3$, $u_4^{min} = -6N$, $u_4^{max} = 6N$, $\omega_{i,j}^{\Delta u} = 0.1\ \forall j = 1, ..., 4$, $i = 0, ..., N_L - 1$. Для выходных переменных устанавливаем нижнюю границу $y_5^{min} = 0$ для переменной z, верхнюю и нижнюю границу $y_{7,9}^{max} = -y_{7,9}^{min} = \frac{\pi}{12}$ для крена и угла наклона. Веса на выходные переменные равны $\omega_j^y = 1$, $j \in \{1, 2, 3, 4\}$ для $x, \dot{x}, y, \dot{y}$ соответственно, и $\omega_j^y = 10$ для всех оставшихся. Горизонт прогноза (prediction horizon) $N_L = 20$, горизонт управления (control horizon) $N_{Lu} = 3$. Период квантования $T_s = 0.05s$. Оставшиеся параметры $V^{y,min}$, $V^{y,max}$, ρ_ϵ устанавливаются по умолчанию с помощью Model Predictive Control Toolbox.

Симуляция нелинейной системы квадрокоптера (3.6) - (3.11) проводилась под эффектом MPC-регулятора (3.14) - (3.17) с помощью Simulink и Model Predictive Control Toolbox. Дополнительные блоки на Рис. 3.2 были добавлены для вывода результатов.

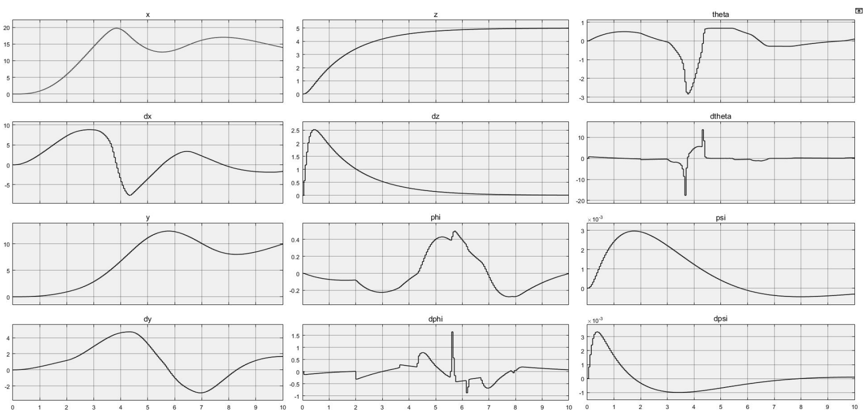

Рис. 3.3: Графики значений выходных данных

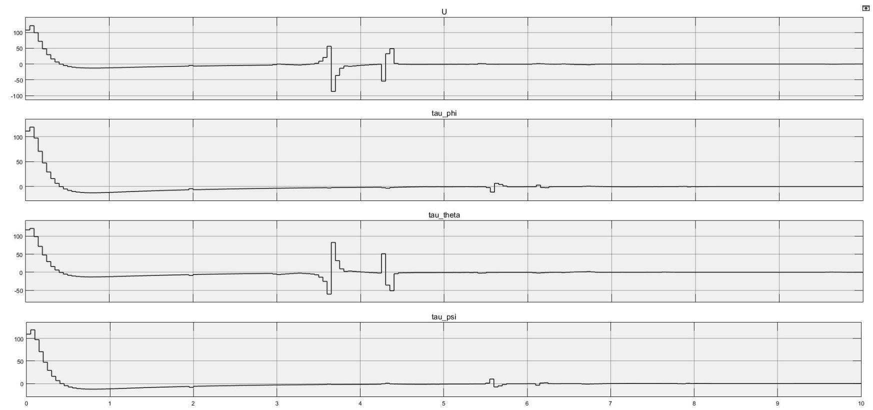

Рис. 3.4: Графики значений входных данных

На Рис. 3.3 показаны результаты симуляции, полученные за счет отслеживания выходных данных. Начальным значением состояния системы являлся нулевой вектор, а ограничениями на положения x, y, z были значения, отклоняющиеся от желаемых на ± 2. К концу симуляции необходимо было достичь определенных значений положения квадрокоптера: $x = 15,\ y = 10,\ z = 5$, что можно увидеть на Рис. 3.3. На Рис. 3.4 показаны значения результирующей силы и трех крутящих моментов. Необходимо отметить, что u в начальный момент времени не равняется нулю, что связано с задачей удержать квадрокоптер в состоянии полета.

3.3 Выводы

В данной курсовой работе происходило ознакомление с базовыми принципами и алгоритмами управления по прогнозирующей модели, задачами оптимального управления и некоторыми численными методами их решения, обучением с подкреплением, методами построения MPC-регулятора на основе методов машинного обучения. Был построен MPC-регулятор для задачи управления квадрокоптером, была проведена его настройка и были освещены результаты проведения симуляции.

ЗАКЛЮЧЕНИЕ

В данной курсовой работе рассмотрена задача построения MPC-регулятора, необходимого для управления квадрокоптером. Для выполнения этой задачи была построена замкнутая модель, состоящая из дискретной линейной системы уравнений, полученной после линеаризации нелинейной динамики квадрокоптера, MPC-регулятора и устройств вывода информации о результатах вычисления управляющей последовательности и вектора состояний системы. Модель была построена с помощью Simulink и Matlab. Была проведена симуляция полученой системы, результаты которой проиллюстрированы на графиках. Задача, поставленная перед MPC-регулятором и квадрокоптером, была выполнена с достаточно большой точностью. Результат можно уточнить за счет сужения ограничений для терминального состояния системы.

Дальнейшие исследования в этой области могут включать в себя: ускорение реакции MPC-регулятора, построение более детальной модели (построение точной модели квадрокоптера), использование улучшенных алгоритмов (например, fast MPC, где вычисления вынесены оффлайн и регулятор будет служить в этом случае как таблица ссылок на результаты вычислений).

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1 Grune, L. Nonlinear model predictive control / L. Grune, J. Pannek // Springer London. - 2011

2 Rawlings, J.B. Model Predictive Control: Theory and Design / J.B. Rawlings, D.Q. Mayne // Madison: Nob Hill Publishing. - 2009

3 Islam, M. Dynamics and control of quadcopter using linear model predictive control approach / M. Islam, M. Okasha, M.M. Idres // AEROS Conference: Materials Science and Engineering. - 2017

4 Rodriguez, J.V. Implementation and comparison of linearization-based and backstepping controllers for quadcopters / J.V. Rodriguez // Eindhoven. - 2017

5 Bangura, M. Real-time Model Predictive Control for Quadrotors / M. Bangura, R. Mahony // 19th IFAC World Congress, Cape Town, South Africa. - 2014

6 Kurak, S. Control and Estimation of a Quadcopter Dynamical Model / S. Kurak, M. Hodzic // Periodicals of Engineering and Natural Sciences, Vol.6, No.1, March 2018. - pp. 63-75

7 Sebatino, F. Quadrotor control: modeling, nonlinear control design and simulation / F. Sebatino // KTH Electrical Engineering. - 2015

8 Neff, A. Linear and Non-Linear Control of a Quadrotor UAV / A. Neff // Clemson University. - 2007

9 Павловец, М.Е. Применение методов машинного обучения в системах управления по прогнозирующей модели / М.Е. Павловец, Н.М. Дмитрук // XIX Международная научная конференция по дифференциальным уравнениям "Еругинские чтения - 2019": материалы международной научной конференции. Могилев, 14-17 мая 2019 г. - С. 127-129

10 Тятюшин, А.И. Методы оптимизации и программная система для решения прикладных задач оптимального управления / А.И. Тятюшкин, О.В. Моржин // Иркутский государственный университет путей сообщения.

11 Орлов, Ю.В. Численные методы задач оптимального обобщенного управления / Ю.В. Орлов, Д.Д. Разумовский // Автомат. и телемех., 1993, выпуск 5, 44-51.