

Университет ИТМО

Реферат

по теме вычислительно эффективные алгоритмы решения
прямой и обратной задачи динамики

Выполнил: студент гр. Р4135, Артемов К.
Проверил: Колюбин С.А.

Санкт-Петербург, 2016 г.

Содержание

1	Введение	3
2	Мотивация	4
3	Общие методы решения задач	6
3.1	Уравнения движения	6
3.2	Динамические модели	7
3.3	Алгоритмы	7
4	Работы по алгоритмам для расчета динамики . .	8
5	Открытые проблемы в области алгоритмов для расчета динамики	10
6	Список литературы	11

1 Введение

Первые работы в области робототехнических систем были написаны более двух десятков лет тому назад. Тем не менее, как в прошлом веке, так и сейчас сообщество робототехники сосредоточено на проблеме вычислительной эффективности. Многие из наиболее эффективных алгоритмов использующиеся сейчас в динамике, которые применимы к широкому классу механизмов, были разработаны учеными очень тесно связанными с роботами.

Наряду с тем, что эффективность алгоритмов имеет несомненно важное значение для моделирования и управления все более сложными механизмами, работающими на более высоких скоростях, другие аспекты динамики также играют важную роль. Алгоритмы формулируются в соответствии с уравнениями движения, что способствует более простой разработке и реализации. Кроме того, важно, чтобы алгоритмы могли решать задачи для как можно более общего случая устройств с разной геометрией и структурой.

2 Мотивация

В принципе, решение задач динамики для роботов не представляет особых трудностей - робототехническое устройство это просто система твердых тел, и как получить уравнения движения таких систем известно достаточно давно. Но на практике не все так просто и встает вопрос о поиске способа получения динамики робота в реальном времени, что приводит к необходимости использовать вычислительно эффективные алгоритмы.

В простейшей форме, уравнение движение для робототехнического механизма может быть представлено вектором дифференциальных уравнений:

$$\ddot{q} = \phi(q, \dot{q}, \tau)$$

где q - вектор обобщенных координат, \dot{q} и \ddot{q} производные по времени, и τ - вектор сил. Решая такое уравнение относительно q необходимо рассчитать значение функции ϕ и выполнить операции интегрирование. Для систем твердых тел сложных робототехнических механизмов основной проблемой является поиск значения функции ϕ .

Для не сложной системы твердых тел, наиболее простым будет выразить уравнения функции для каждого из параметров, результатом чего станет набор простых функций для каждой из степеней свободы. Далее, вычисляя их, получим числовые значения.

К сожалению, такой подход не работает для более сложных систем, в которых число параметров стремительно растет с увеличением количество тел в системе. Что требует применения специальных методов и алгоритмов. Все динамические алгоритмы для сложным многосвязных тел оценивают уравнения движения поэтапно. На каждом из этапов получают значения, которые используются на следующем этапе вычислений и так далее.

Динамика робототехнических систем представляет собой отношения между действующими на механизм силами и ускорениями, которые они производят. Как правило, робототехнические механизмы моделируются как система твердых тел, в таком случае динамика такой системы также рассматривается как динамика твердого тела. Существуют две основные задачи динамики:

прямая задача – даны силы, найти ускорения;

обратная задача – даны ускорения, найти силы.

Прямая задача в основном используется для моделирования. Обратная задача более распространена, например, для: управления движением и силой робота в режиме реального времени; планирования траекторий и оптимизации; при разработке роботов; а также, как один из элементов алгоритмов прямой задачи.

Также задачами динамики являются:

вычисление коэффициентов управления движением;

определение параметров инерции – оценка параметров инерции механизмов робота посредством измерения их динамического поведения

гибридная динамика: даны силы в некоторых сочленениях и ускорения в других, найти неизвестные силы и ускорения.

Для определения всех необходимых значений параметров в режиме реального времени стоит задача вычисления их, тут-то и пригождаются вычислительно эффективные алгоритмы.

3 Общие методы решения задач

3.1 Уравнения движения

Уравнения движения для робототехнической системы можно записать, как:

$$\tau = H(q)\ddot{q} + c(q, \dot{q}, f_{ext})$$

где q , \dot{q} , и \ddot{q} – векторы позиции сочленения, его скорости, ускорения и силы, соответственно, каждый из которых имеет размерность n , где n – количество независимых сочленений в механизме. f_{ext} – внешняя сила действующая на робота, соответствует взаимодействию робота с окружающей средой. Часто, если робот имеет только одну точку взаимодействия с внешней средой то, f_{ext} – в пространственном векторе выражает действие внешней силы на схват робота. Если таких точек много, то рассматривается робот с множеством схватов (как например руки и ноги у роботов-гуманоидов). H – матрица инерции робота, размерности $n \times n$, симметричная. c – сила смещения, значение силы, которая должна прикладываться к системе в случае отсутствия ускорения.

Это уравнение может быть полезно для вычислений. Однако, есть два других способа записи уравнения движения.

- joint-space formulation

$$H(q)\ddot{q} + C(q, \dot{q})\dot{q} + \tau_g(q) = \tau$$

- operational-space formulation

$$\Lambda(x)v + \mu(x, v) + \rho(x) = f$$

где x – положение схвата, v – пространственный вектор скорости схвата, f – пространственный вектор действующей силы на схват и Λ и ρ – коэффициенты уравнения движения. Λ это матрица инерции в операционном пространстве, μ – содержит кориолисовы и центробежные составляющие; и ρ – вектор грави-

тации, f - сумма внешних сил, действующих на схват. Эти уравнения показывают функциональную зависимость: H - функция от q . Λ функция от x . Однако, эти зависимости часто отбрасывают, для более простого понимания. x - это вектор координат в операционном пространстве, v и f пространственные вектора, обозначающие скорости схвата и действие внешних сил на него.

3.2 Динамические модели

Строго говоря, коэффициенты уравнений зависят не только от q, \dot{q} и f_{ext} , но также от динамической модели робототехнической системы. Т.е. описание механизма отдельно для каждой из его части: звенья, сочленения и их характеризующие их параметры. Динамическая модель состоит из: кинематической модели множества параметров инерции. Для описание инерции твердого тела нужно десять параметров инерции: масса, расположение центра масс, шесть вращательных параметров инерции. Однако, когда тела соединяют друг с другом их степени свободы ограничиваются и некоторые из этих параметров могут не оказывать влияния на поведение системы. Поэтому при составлении динамической модели робототехнического механизма от кинематической модели и измерения ее динамических характеристик, процедура сводится к идентификации значений множества основных параметров инерции, которые поддаются наблюдению в полученной системе.

3.3 Алгоритмы

Наиболее эффективными алгоритмами для расчет динамических параметром робототехнических систем используются следующие три:

1. рекурсивный алгоритм Ньютона-Эйлера (RNEA). Алгоритм представляет собой решение обратной задачи динамики и

имеет вычислительную сложность $O(n)$. В зависимости от параметров запуска может быть использован для расчета: вектора гравитации, кориолисовых/центробежных сил, матрицы инерции и обобщенного момента.

2. articulated-body algorithm (ABA). Решает прямую задачу динамики. Вычислительная сложность $O(n)$.
3. composite-rigid-body algorithm (CRBA). Предназначен для расчета матрицы инерции в для пространства обобщенных координат. Вычислительная сложность $O(n^2)$. При комбинации его с алгоритмом RNEA, решается прямая задача динамики с вычислительной сложностью $O(n^3)$. Хотя эта комбинация получается гораздо медленнее, чем алгоритм ABA для больших значений, он получается таким же эффективным при значениях n порядка шести.

По приблизительным данным: RNEA требует около 200 вычислений с плавающей точкой для каждого тела, ABA требует около 450 и CRBA - около $16n^2$ вычислений для матрицы инерции размером $n \times n$.

4 Работы по алгоритмам для расчета динамики

Классический подход к выражению уравнений движения был основан на методе Эйлера-Лагранжа. Разрабатываемые алгоритмы использовали динамику Лагранжа, сложность вычисления которой порядка $O(n^4)$, которые нужно адаптировать для систем реального времени. Но для некоторых задач были разработать алгоритмы более низкого порядка:

1. прямая задача динамики, которая заключается по известным моментам/силам в сочленениях найти траекторию движения манипулятора (позиция, скорость и ускорение),

2. обратная задача динамики – для заданного уравнения движения найти моменты/силы в сочленениях,
3. определение матрицы инерции.

Прямая задача решается для моделирования, обратная для управления, матрица инерции находится для анализа, и является неотъемлемой частью многих подзадач прямой задачи динамики.

Первые исследователи разработали $O(n)$ алгоритм для решения обратной задачи динамики используя метод Ньютона-Эйлера (NE). Степаненко и Вукобратович разработали рекурсивный метод NE для вычисления динамических параметров конечностей человека. Затем Орин усовершенствовал этот метод и применил его для управления ногами шагающего робота в режиме реального времени. Лух, Уолкер и Пол разработали очень эффективный рекурсивный NE алгоритм (RNEA), который позволяет рассчитывать динамику робота с большим количеством звеньев. Холлербах разработал $O(n)$ рекурсивный алгоритм используя метод Лагранжа, но обнаружил, что он менее эффективен по сравнению с RNEA, так как требовалось гораздо большее число умножений и сложений/вычитаний для одной и той же задачи. Далее, в результате работы Балафотиса и Голденберга алгоритм удалось сделать в 1.7 раза быстрее предыдущих его реализаций (для роботов с 6-ю степенями свободы). Уолкер и Орин используя RNEA для решения обратной задачи динамики как основополагающий применили его для решения прямой задачи. Получившийся алгоритм впоследствии получил название composite-rigid-body algorithm (CRBA) благодаря Физерстону, плотно работающему со сложными манипуляционными цепочками, состоящие из множества тел. Еще один известный алгоритм $O(n)$ для решения прямой задачи динамики был разработан Верещагиным. В этом алгоритме используется рекурсивная формула уравнения движения, применяется для

неразветвленных цепей с вращательными и призматическими сочленениями. Этот алгоритм напоминает Articular-Body алгоритм. Армстронг разработал алгоритм $O(n)$ для механизмов со сферическими сочленениями, а затем Фетерсон разработал уже и сам АВА. Первая версия этого алгоритма применима к манипуляторам с одной степенью свободы, а вторая включала основные типы сочленений и была быстрее. С точки зрения общего числа арифметических операций АВА был более эффективен, нежели CRBA для $n > 9$.

5 Открытые проблемы в области алгоритмов для расчета динамики

6 Список литературы

1. Featherstone, R. (1987). Robot Dynamics Algorithms
2. Siciliano, B., and Khatib, O. (eds.) (2008). Springer Handbook of Robotics
3. Featherstone, Orin Robot Dynamics: Equations and Algorithms