

Федеральное государственное автономное образовательное учреждение
высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»
Физтех-школа Прикладной Математики и Информатики
Центр обучения проектированию и разработке игр

Направление подготовки / специальность: 09.03.01 Информатика и вычислительная техника

Направленность (профиль) подготовки: Компьютерные науки и инженерия

УЛУЧШЕНИЕ ГЕНЕРАЦИИ ДВИЖЕНИЯ РЭГДОЛА

(бакалаврская работа)

Студент:

Москалев Никита Евгеньевич

(подпись студента)

Научный руководитель:

Козак Роман Александрович

(подпись научного руководителя)

Москва 2024

Аннотация

Содержание

1	Введение	4
1.1	Модель персонажа	4
1.1.1	Кинематическое дерево	5
1.1.2	Обобщенные координаты	6
1.1.3	Уравнение динамики	6
1.1.4	Импульс и момент импульса	7
2	Система управления персонажем	9
2.1	Оптимизация	9
2.1.1	Следование за опорной анимацией	10
2.1.2	Управление положением центра масс	10
2.1.3	Управление положением центра давления	11
2.1.4	Положением точек контакта с поверхностью	11
2.1.5	Направление силы реакции опоры	11

1. Введение

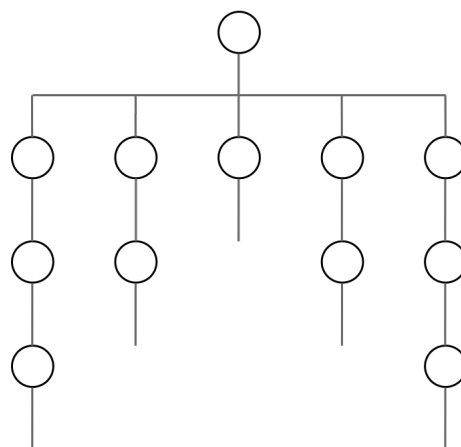
Одна из основных задач, стоящих перед разработчиками и исследователями в области анимации, – это воспроизведение разнообразия движений человека, с целью создания персонажей, способных реалистично взаимодействовать с игровым миром. Несмотря на сложности, связанные с недостаточным пониманием всех тонкостей движения человека, множество подходов к решению этой задачи было предложено за последние десятилетия. Существующие подходы можно разделить на две группы: основанные на данных и основанные на физическом моделировании. Первая группа использует большое количество предварительно записанных движений человека, для создания необходимых анимаций персонажа. Такие подходы демонстрируют хорошие результаты, поскольку полагаются на данные собранные в реальном мире, однако плохо применимы в случае когда требуемое движение не стандартно и не похоже на имеющиеся. Вторая группа управляет динамикой физической модели персонажа. В таком случае обеспечивается физическая корректность и появляется возможность взаимодействия с виртуальной средой.

Широко распространенная проблема управления – это поддержание баланса при движении, то есть предотвращение неконтролируемого падения персонажа. Особенно ярко проблема проявляется, когда персонаж находится в присутствии внешних возмущений или на неровной поверхности. Кроме того, проблема сильно варьирует в зависимости от вида производимого движения. Поэтому выделяют две категории баланса: статический, для движений на месте, и динамический.

В данной работе предлагается и анализируется способ поддержания статического баланса персонажа, следующего опорной анимации на основе локальной оптимизации и индикаторов баланса, описанных в [1]. В каждый момент времени решается задача квадратичного программирования, которая оптимизирует значения целевых функций, отвечающих за баланс и следование за анимацией. После чего персонаж приводится в движение в соответствии с результатами оптимизации. Более того, при изменении целевых функций описываемый способ может быть адаптирован для других проблем управления.

1.1. Модель персонажа

Перед тем как перейти к описанию разработанной системы, построим физическую модель персонажа и сформулируем уравнения, описывающие ее.



1.1.1. Кинематическое дерево

Кинематическое дерево – это система из n твердых тел, соединенных между собой m шарнирами. Каждый из шарниров, кроме корневого, ограничивает относительное движение тел, которые он соединяет. Например, призматический шарнир оставляет только поступательное движение вдоль выбранной оси. Корневой шарнир, в свою очередь, определяет возможность системы перемещаться в пространстве и бывает двух видов: плавающий, то есть не накладывающий ограничений, и фиксирующий.

Для того чтобы приводить в движение отдельные тела, некоторые шарниры могут быть снабжены приводами, которые генерируют необходимые силы и моменты сил. В таком случае шарнир называется активным, иначе, соответственно, неактивным. Отметим, что корневой шарнир обычно остается без привода. Таким образом за движение системы как целого отвечает сила трения, что сохраняет физическую корректность, но сильно усложняет управление кинематическим деревом.

В данной работе персонаж моделируется как кинематическое дерево, все шарниры которого имеют привод, причем силы и моменты сил, генерируемые в корневом шарнире, минимизируются в оптимизаторе. Такая модель в большинстве ситуаций эквивалентна кинема-

тическому дереву с неактивным корневым шарниром, однако оставляет возможность нарушить физическую корректность при необходимости. На рисунках 1 и 2 изображен пример трехмерной модели персонажа и соответствующего кинематического дерева.

1.1.2. Обобщенные координаты

При работе с кинематическим деревом важную роль играет способ, выбранный для описания положения и ориентации тел в пространстве, поскольку он во многом определяет простоту, устойчивость и вычислительную сложность моделирования. Основные способы – это максимальные координаты и обобщенные координаты.

Максимальные координаты описывают тела по отдельности, используя по 6 чисел на каждое, а ограничения, накладываемые шарнирами, учитывают при решении уравнения динамики. Такой способ позволяет использовать существующие системы физического моделирования, но страдает от ошибок работы с вещественными числами, которые приводят к тому, что тела открепляются друг от друга.

Обобщенные координаты, напротив, учитывают связи между телами. Например, для кинематического дерева, состоящего из двух тел, соединенных вращательным шарниром, используется 7 чисел, первые 6 из которых описывают положение и ориентацию одного из тел, а оставшееся – угол поворота вокруг оси шарнира. Такой способ минимизирует количество используемых чисел и неявно учитывает ограничения.

В данной работе используются обобщенные координаты, скорости и ускорения. Отметим, что обобщенные скорости позволяют выразить скорость любого шарнира. Для этого необходимо умножить их на якобиан шарнира. После чего можно получить скорость любой точки твердого тела, прикрепленного к шарниру, зная ее положение.

1.1.3. Уравнение динамики

Уравнение динамики кинематического дерева, может быть выведено из принципа наименьшего действия, и имеет вид

$$H\ddot{q} + C(q, \dot{q}) + G(q) = u + J^T f, \quad (1.1)$$

где q, \dot{q}, \ddot{q} – обобщенные координаты, скорости и ускорения, H – матрица инерции, C – центробежная и кориолисова силы, G – сила тяготения, u – силы и моменты сил, генерируемые приводами, J – якобиан, и f – внешние силы.

В качестве неизвестной в уравнении 1.1 может выступать \ddot{q} или u . Алгоритмы, которые находят \ddot{q} , называются алгоритмами прямой динамики, а те, которые находят u , – обратной.

Для приведения кинематического дерева в движение необходимо использование алгоритма прямой динамики. В данной работе используется алгоритм articulated rigid body, разработанный и описанный в [2], который имеет меньшую вычислительную сложность в сравнении с остальными.

1.1.4. Импульс и момент импульса

В разработке систем управления персонажем активно используются такие величины как импульс и момент импульса кинематического дерева, поскольку их производные тесно связаны с положениями центра масс и центра давления (уравнения 1.9, 1.10 и 1.11), которые, в свою очередь, являются индикаторами баланса.

Напомним, что центр давления – это точка, где можно приложить силу реакции опоры, то есть результирующую сил нормальной реакции опоры и сил трения, так чтобы момент этих сил относительно центра масс не изменился.

В [3] показана связь импульса и момента импульса кинематического дерева, выраженных в неподвижной системе отсчета, расположенной в центре масс, с обобщенными скоростями, имеющая следующий вид

$$\begin{bmatrix} P \\ L \end{bmatrix} = A\dot{q}, \quad (1.2)$$

где P – импульс, L – момент импульса, а A – это центроидальная матрица.

Центроидальная матрица, как и матрицей инерции, является фундаментальной характеристикой кинематического дерева, которая зависит только от массы, инерции и значения обобщенных координат твердых тел, составляющих его.

При разделении центроидальной матрицы на две уравнение 1.2 принимает вид

$$P = A_P\dot{q}, \quad (1.3)$$

$$L = A_L\dot{q}. \quad (1.4)$$

При дифференцировании уравнения 1.3 и 1.4 принимают вид

$$\dot{P} = \dot{A}_P\dot{q} + A_P\ddot{q}, \quad (1.5)$$

$$\dot{L} = \dot{A}_L\dot{q} + A_L\ddot{q}. \quad (1.6)$$

Полученные уравнения 1.5 и 1.6 позволяют вычислить обобщенные ускорения необходимые чтобы производные импульса и момента импульса приняли требуемые значения, а следовательно и положения центра масс и центра давления.

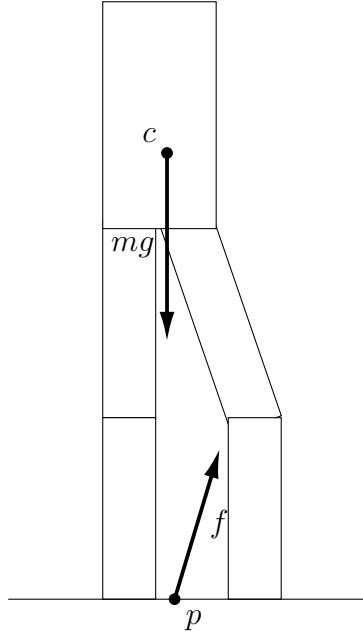


Рисунок 3: Силы

Рассмотрим силы, действующие на кинематическое дерево (рисунок 3). Запишем второй закон Ньютона и основное уравнение вращательной динамики

$$\dot{P} = mg + f, \quad (1.7)$$

$$\dot{L} = (p - c) \times f, \quad (1.8)$$

где c – центр масс, p – центр давления, m – суммарная масса, и f – результирующая сил нормальной реакции опоры и сил трения.

Исключая f из уравнений 1.7 и 1.8, получим

$$\dot{L} = (p - c) \times (\dot{P} - mg). \quad (1.9)$$

Теперь, запишем определение импульса кинематического дерева, рассматривая его как систему твердых тел

$$P = \sum_{i=1}^n m_i \dot{x}_i,$$

где x_i , m_i – положения и массы. Используя следующую цепочку равенств

$$\sum_{i=1}^n m_i \dot{x}_i = \frac{d}{dt} \left(\sum m_i x_i \right) = \frac{d}{dt} (mc) = m\dot{c},$$

преобразуем уравнение к виду

$$P = m\dot{c}, \quad (1.10)$$

а дифференцированием к виду

$$\dot{P} = m\ddot{c}. \quad (1.11)$$

2. Система управления персонажем

Работа системы управления персонажем состоит из двух стадий: оптимизации и прямой динамики, повторяемых в цикле. Во время оптимизации вычисляются силы и моменты сил, которые должны сгенерировать приводы, и силы реакции опоры. Результаты оптимизации передаются в алгоритм прямой динамики, который находит ускорения для дальнейшего интегрирования.

Внешние возмущения, если присутствуют, также передаются в алгоритм прямой динамики. Ошибки, которые они вносят в движение, компенсируются оптимизатором на следующих циклах.

На рисунке 4 схематично изображена работа системы.

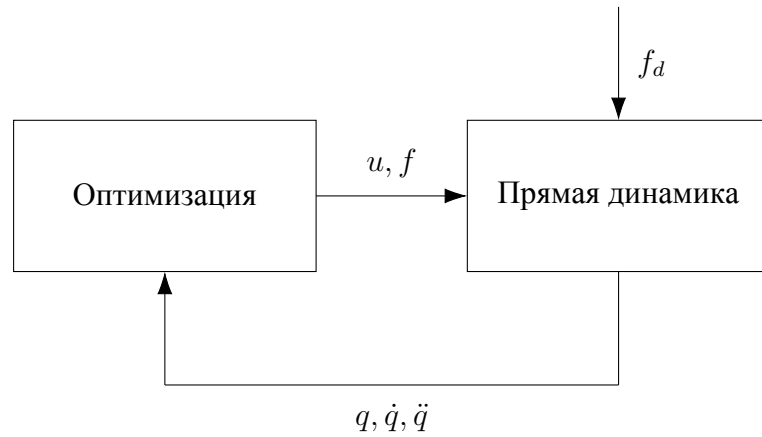


Рисунок 4: Работа системы управления

Отметим, что похожая схема работы описана в [4]. Однако она имеет другую формулировку задачи оптимизации, что является важным отличием, делающим систему, предложенную в данной работе, более надежной.

2.1. Оптимизация

Задача оптимизации имеет следующий вид

$$\min_{\ddot{q}, u, f} \omega_t h_t(\ddot{q}) + \omega_c h_c(\ddot{q}) + \omega_p h_p(\ddot{q}) + \omega u^T \begin{bmatrix} I_6 & O \\ O & O \end{bmatrix} u$$

$$s.t. H\ddot{q} + C(q, \dot{q}) + G(q) = u + J^T f \quad (2.1)$$

$$J_{sup}\ddot{q} + \dot{J}_{sup}\dot{q} = 0 \quad (2.2)$$

$$0 \leq n(f) \quad (2.3)$$

$$\tau(f) \leq \mu n(f) \quad (2.4)$$

Целевая функция оптимизации состоит из четырех слагаемых. Первые три из них – h_t , h_c и h_p – отвечают за следование опорной анимации, за положения центра масс и за положения центра давления соответственно. Последнее стремиться минимизировать силы и моменты сил, генерируемые в корневом шарнире. Веса w_t , w_c , w_p и w – позволяют регулировать значимость вклада каждого из слагаемых.

Уравнение 2.1 согласует результаты оптимизации с уравнением динамики кинематического дерева. Уравнение 2.2 оставляет точки контакта с поверхностью неподвижными. Неравенства 2.3 и 2.4 отвечают за направление силы реакции опоры.

Функции h_t , h_c и h_p формулируются так, чтобы получившаяся задача оптимизации была квадратичной. Такое ограничение дает предсказуемое время работы оптимизатора и стабильные результаты.

2.1.1. Следование за опорной анимацией

Функция h_t должна как можно точнее приближать получаемое движение к опорной анимации. На первый взгляд, для этого, достаточно определить h_t таким образом, чтобы в результате оптимизации ускорение совпадало с тем, что взято из опорной анимации. Однако, при таком определении, в случае отклонения получаемого движения от опорной анимации оно не будет скорректировано обратно. Поэтому h_t будет определена следующим образом

$$h_t(\ddot{q}) = \|\ddot{q} + s_t(q - q_{ref}) + d_t(\dot{q} - \dot{q}_{ref}) - \ddot{q}_{ref}\|_2^2,$$

где s_t и d_t – коэффициенты, а q_{ref} , \dot{q}_{ref} и \ddot{q}_{ref} – это положение, скорость и ускорение взятые из опорной анимации.

Выражение $s_t(q - q_{ref}) + d_t(\dot{q} - \dot{q}_{ref})$ использованное в определении h_t называется пропорционально-дифференцирующий регулятор, а коэффициенты s_t и d_t – пропорциональный и дифференциальный соответственно. Использование этого регулятора позволяет преодолеть проблему, описанную выше, а также смягчить ошибки связанные с работой с числами с плавающей точкой.

2.1.2. Управление положением центра масс

Функция h_c должна поддерживать проекцию положения центра масс на поверхность внутри выпуклой оболочки, образованной точками контакта с поверхностью, поскольку в противном случае персонаж перевернется.

Для этого h_c определяется таким образом, чтобы в результате оптимизации текущее положение центра масс двигалось в направлении, предварительно выбранного, опорного по-

ложения. А точнее, ускорение центра масс принимало следующий вид

$$\ddot{c}_{des} = -s_c(c - c_{ref}) - d_c\dot{c}, \quad (2.6)$$

где s_t и d_t – коэффициенты, а c_{ref} – опорное положение.

Умножением обеих частей на массу, уравнение 2.6 принимает вид

$$m\ddot{c}_{des} = -s_cm(c - c_{ref}) - d_cm\dot{c}. \quad (2.7)$$

Заменой $m\ddot{c}_{des}$ и $m\dot{c}$ через уравнения 1.10 и 1.11, уравнение 2.7 принимает вид

$$\dot{P}_{des} = -s_cm(c - c_{ref}) - d_cP. \quad (2.8)$$

Таким образом управление положением центра масс сводится к управлению величиной импульса, что сделать не сложно, поскольку есть зависимость между ним и обобщенными координатами, скоростями и ускорениями, выраженная в виде уравнения 1.5. В итоге h_c определяется следующим образом

$$h_c(\ddot{q}) = \|P - P_{des}\|_2^2 = \|A_P\ddot{q} - \dot{A}_P\dot{q} - P_{des}\|_2^2.$$

2.1.3. Управление положением центра давления

2.1.4. Положением точек контакта с поверхностью

2.1.5. Направление силы реакции опоры

Список литературы

1. *Macchietto A., Zordan V., Shelton C. R.* Momentum control for balance // ACM SIGGRAPH 2009 Papers. — 2009. — DOI: [10.1145/1576246.1531386](https://doi.org/10.1145/1576246.1531386).
2. *Featherstone R.* Rigid Body Dynamics Algorithms. — Springer, 2007. — DOI: [10.1007/978-1-4899-7560-7](https://doi.org/10.1007/978-1-4899-7560-7).
3. *Orin D. E., Goswami A.* Centroidal Momentum Matrix of a humanoid robot: Structure and properties // 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems. — 2008. — DOI: [10.1109/IROS.2008.4650772](https://doi.org/10.1109/IROS.2008.4650772).
4. *Abe Y., Silva M. da, Popović J.* Multiobjective control with frictional contacts // Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation. — 2007. — DOI: [10.5555/1272690.1272724](https://doi.org/10.5555/1272690.1272724).