

РАСЧЕТ ОБЛАСТЕЙ ПЕРЕСЕЧЕНИЯ ПОВЕРХНОСТЕЙ ЗАХВАТНЫХ УСТРОЙСТВ МАНИПУЛЯТОРОВ И ДЕФОРМИРУЕМЫХ ОБЪЕКТОВ ПРИ ПЛАНИРОВАНИИ И МОДЕЛИРОВАНИИ ЗАХВАТА

А.Г. Лесков¹

agleskov@rambler.ru

Е.В. Селиверстова^{1,2}

feoktistovaev@mail.ru

¹Дмитровский филиал МГТУ им. Н.Э. Баумана, пос. Орево, Дмитровский р-н, Московская обл., Российская Федерация

²МГТУ им. Н.Э. Баумана, Москва, Российская Федерация

Аннотация

Выполнен анализ методов обнаружения и расчета областей пересечения поверхностей хватных устройств манипуляторов и объектов манипулирования при решении задач моделирования и планирования захвата деформируемых объектов. Предложен алгоритм обнаружения и расчета областей пересечения, включающий в себя широкую и узкую фазы. Алгоритм широкой фазы основан на методе ограничивающих параллелепипедов и усовершенствован благодаря введению оригинального алгоритма обнаружения областей потенциального взаимодействия хватных устройств манипуляторов и объектов манипулирования. Алгоритм узкой фазы является оригинальным. В его основе — алгоритм поиска ближайших соседей с учетом направления движения взаимодействующих тел. Разработана компьютерная программа и приведены результаты применения предложенных алгоритмов для моделирования захвата объекта трехпалой кистью Schunk SDH

Ключевые слова

Захват, деформируемый объект, моделирование, пересечение полигональных моделей, Oriented Bounding Box, VP-дерево

Поступила в редакцию 27.05.2016

© МГТУ им. Н.Э. Баумана, 2016

Одна из наиболее распространенных манипуляционных операций — это автоматический захват объекта манипулирования (ОМ). Известны многочисленные источники [1, 2], в которых рассматриваются вопросы захвата ОМ, представляющих собой жесткие тела. В отличие от захвата жестких тел, захват деформируемых тел является менее изученным.

В процессе захвата деформируемого тела элемент хватного устройства манипулятора (ЗУМ) деформирует ОМ. В результате формируется пара пятен контакта, принадлежащих поверхностям ОМ и ЗУМ. Эти пятна имеют одинаковую форму. Эпюры сил, действующих на ОМ и ЗУМ, зеркально симметричны относительно поверхности пятна контакта. Если звено ЗУМ недеформируемое, то геометрия пятен контакта идентична геометрии области поверхности звена, которая находится в контакте с ОМ. В противном случае она зависит от формы ОМ и звена, а также от их упругих свойств.

При контакте деформируемых элементов ЗУМ с деформируемыми ОМ происходит изменение первоначальной формы контактирующих элементов. В настоящей работе контакт между деформируемыми элементами определяется как геометрическое пересечение поверхностей первоначальных (еще не деформированных) форм контактирующих элементов. Такое рассмотрение геометрически имитирует контактные деформации. Отметим, что приведенное понятие пересечения поверхностей является условным, его использование — общепринятым приемом при моделировании захвата тел [3].

Одним из способов синтеза захвата является офлайн-планирование по полигональным моделям ЗУМ и ОМ [1], обеспечивающее предварительную оценку возможных способов захвата. Моделирование происходит путем пошагового изменения положения элементов ЗУМ для перевода его в состояние «схват закрыт».

В настоящей статье рассматривается один из центральных вопросов, возникающих при моделировании захвата упруго деформируемых объектов, — обнаружение пересечений поверхностей ЗУМ и ОМ. При решении этой задачи необходимо установить факт пересечения и рассчитать его параметры: область пересечения, которая представляет собой фигуру, образованную в результате конъюнкции фигур ЗУМ и ОМ; глубину проникания ЗУМ в ОМ при захвате, т. е. минимальное расстояние, на которое нужно развести эти объекты, чтобы их поверхности перестали пересекаться; вектор, вдоль которого отсчитывается глубина проникания.

При моделировании захвата упругих тел перечисленные параметры используются для определения геометрии пятен контакта и эпюр нагрузки, которые участвуют в расчете сил контактного взаимодействия между ЗУМ и ОМ.

Методы и алгоритмы обнаружения пересечения объектов. При решении задачи обнаружения пересечения поверхности ЗУМ и ОМ представляются в виде многогранников (полигональных моделей). Полигональные модели задаются наборами вершин многогранников и соединяющих их ребер (рис. 1, а, в). Захватное устройство представляется в виде набора полигональных моделей, каждая из которых соответствует одному элементу разветвленной кинематической цепи схвата (рис. 1, б).

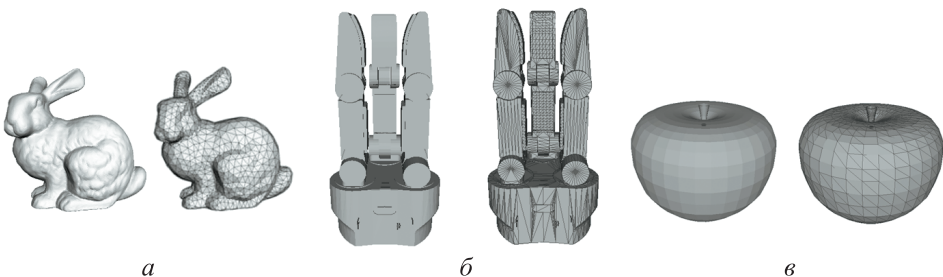


Рис. 1. Полигональные модели ОМ (а, в) и захватного устройства Schunk SDH (б)

Сложность детектирования пересечений ЗУМ и ОМ при полигональном моделировании заключается в том, что для обеспечения приемлемой точности необходимо вводить в рассмотрение подробные модели поверхностей объектов. Точность моделирования поверхности объекта тем выше, чем больше входящих в модель полигонов. Следовательно, для расчета пересечений в приемлемое время потребуются значительные вычислительные ресурсы, пропорциональные числу полигонов и вершин моделей. Необходимый компромисс между точностью и временем вычислений можно найти, учитывая следующий факт.

При выполнении захвата в начальный момент времени контакт между звеньями ЗУМ и ОМ отсутствует — между ними имеется зазор. В процессе смыкания пальцев ЗУМ зазор постепенно уменьшается, затем происходит пересечение поверхностей ЗУМ и ОМ и объект захватывается. Исходя из этого, можно снизить ресурсоемкость вычислений при достаточной точности путем разбиения процедуры обнаружения пересечений на две фазы. В первой фазе моделирования захвата можно использовать «грубые» и менее ресурсоемкие методы обнаружения пересечения. В этой фазе просто устанавливается факт столкновения ЗУМ и ОМ. Более точные методы расчета пересечения поверхностей ЗУМ и ОМ применяются во второй фазе операции — после обнаружения пересечения ЗУМ и ОМ в первой фазе. В литературе эти этапы получили названия широкой (быстрая и грубая оценка) и узкой (более медленная и точная) фаз обнаружения столкновений [4].

Анализ существующих методов и алгоритмов широкой фазы. Наиболее часто используемыми алгоритмами широкой фазы обнаружения пересечений являются методы ограничивающих объемов (ОО) — в виде сферы и параллелепипедов.

В методе ОО элементы конструкции ЗУМ и ОМ полагаются помещенными внутрь других фигур, имеющих более простую форму. Расчет расстояний между ограничивающими объемами ЗУМ и ОМ требует значительно меньших затрат времени. Потеря точности компенсируется при реализации узкой фазы расчета захвата.

Использование в качестве ОО сферы является самым простым, быстрым, но и самым грубым приемом обнаружения столкновений (рис. 2, а). Фигура в виде параллелепипеда более точно, чем сфера, повторяет форму объектов. При этом проверка на пересечение с использованием в качестве ОО параллелепипедов (ОП) выполняется также, как и в случае сферы, достаточно быстро. Поэтому ОП наиболее часто используют для проверки на наличие пересечения.

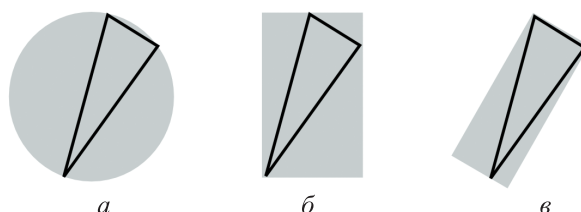


Рис. 2. Ограничивающие объемы

Используют два вида ОП — с привязкой к осям координат (Axis-Aligned Bounding Box, AABB) и ориентируемый ОП (Oriented Bounding Box, OBB). Ребрами, параллельными осям мировой системы координат, задается AABB. При вращении объекта AABB изменяет свои размеры, но всегда остается ориентированным по осям некоторой неподвижной системы координат (СК). Поэтому наличие пересечения проверяется сравнительно просто. Однако использование AABB часто приводит к появлению таких ОО, форма которых значительно отличается от реальных прототипов (рис. 2, б). Это снижает точность результатов.

Ориентируемый ОП (OBB) имеет ребра, изменяющие свою ориентацию в пространстве соответственно ориентации объекта. Соответствующим выбором параметров можно выбрать такой OBB, который плотно облегает объект (рис. 2, в). В отличие от AABB, OBB вращается вместе с объектом и не меняет своих размеров. Однако проверка на пересечение OBB осуществляется сложнее, чем в рассмотренных ранее методах ограничивающей сферы и AABB.

Решение о выборе между AABB или OBB зависит от условий конкретной задачи. Для построения OBB как такового требуется больше вычислительных ресурсов, чем для построения AABB. Параметры OBB рассчитываются один раз в начале операции захвата. В последующем OBB не изменяет свою форму (изменяется только его положение и ориентация в пространстве в соответствии с текущим положением и ориентацией объекта).

Поскольку в общем случае OBB более точно, чем AABB, повторяет форму объекта, точность определения факта пересечения OBB выше, чем AABB. Поэтому, если проверка на пересечение осуществляется между небольшим числом объектов, то целесообразно использовать OBB как более точный. Если же на каждом шаге могут появляться новые объекты, то применяют более быстрый расчет для AABB.

Модернизированный алгоритм широкой фазы. Поскольку поверхности ОО часто значительно выступают за пределы очертаний объектов, пересечение поверхностей ОО не всегда соответствует пересечению поверхностей ЗУМ и ОМ. Факт пересечения можно установить только после перехода к полигональному представлению их поверхностей. Однако, как было отмечено, это требует существенного увеличения объема вычислений. Объем вычислений можно сократить, если каким-либо образом ограничить список полигонов и вершин полигональной модели. Далее предложен алгоритм решения такой задачи.

Введем в рассмотрение некоторые области потенциального взаимодействия (ОПВ), каждая из которых соответствует одному из ОО. В качестве ОО примем OBB. Такой выбор обусловлен относительно небольшим числом участвующих в рассмотрении объектов.

Область потенциального взаимодействия имеет вид параллелепипеда, расположенного внутри соответствующего OBB и ограничивающего пространство проникновения другого OBB. Она строится таким образом, что ее ребра параллельны ребрам OBB, которому она соответствует и внутри которого находится. Область потенциального взаимодействия для ОП ОМ и ОП звена ЗУМ изображены на рис. 3, а зелеными и красными линиями.

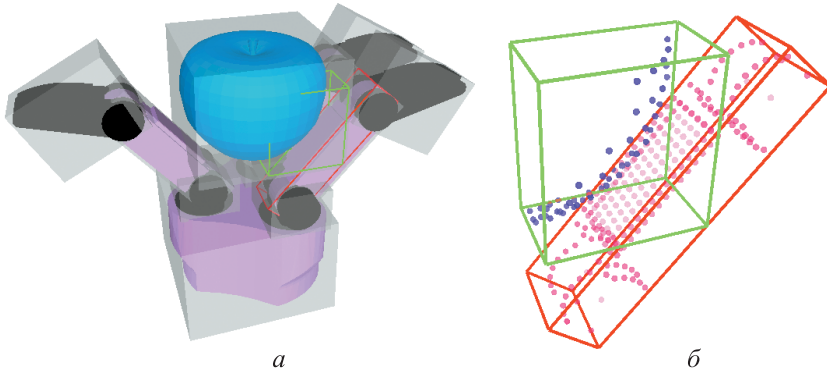


Рис. 3. Области потенциального взаимодействия ОМ с одним из звеньев ЗУМ

На рис. 3, б эти же ОПВ представлены отдельно. Синими точками обозначены вершины ОМ, находящиеся внутри ОПВ, соответствующей ОБВ ОМ, розовыми — вершины звена ЗУМ, расположенные в ОПВ ОБВ звена ЗУ. Видно, что число точек и полигонов, вершинами которых они являются, существенно меньше общего числа вершин и полигонов моделей поверхностей ОМ и ЗУМ.

Очевидно, что пересечение поверхностей ОМ и ЗУМ может происходить только внутри ОПВ. Причем для практики представляют интерес только те случаи, когда в ОПВ находятся одновременно вершины многогранников и ОМ, и ЗУМ.

Подход с использованием ОПВ увеличивает надежность определения факта пересечения, поскольку уменьшается число случаев, когда имеет место пересечение ОП, но пересечение между полигональными моделями отсутствует. Кроме того, использование ОПВ позволяет модернизировать алгоритм широкой фазы, а именно: расчеты в широкой фазе выполняются не только до момента пересечения ОП ОМ и ЗУМ, но и после состоявшегося пересечения этих ОП.

Основанием для перехода к узкой фазе служит факт обнаружения в ОПВ вершин многогранников поверхностей и ОМ, и ЗУМ.

Анализ существующих методов и алгоритмов узкой фазы. В узкой фазе определения пересечения поверхностей объектов с использованием их полигональных моделей чаще всего используются алгоритмы, базирующиеся на методе Gilbert — Johnson — Keerthi (GJK) [5], которые определяют характеристики пересечения полигонов поверхностей путем анализа разности Минковского для векторов, описывающих полигоны, и алгоритмы, опирающиеся на расчет взаимного расположения областей Вороного фрагментов многогранников: алгоритм Lin — Canny [6] и алгоритм V-Clip [7].

В работе [7] приводится сравнение быстродействия алгоритмов GJK, Lin — Canny и V — Clip. Показано, что скорость определения пересечения объектов алгоритма V — Clip значительно больше скорости других алгоритмов. Алгоритм V — Clip используется во многих физических движках — компьютерных программах, моделирующих в виртуальной среде поведение объектов реального мира. В качестве примера можно привести Bullet Physics Library [8] и

Open Dynamics Engine [9]. Но несмотря на явные достоинства, метод V-Clip сложен в реализации, требует предварительного расчета областей Вороного. Результат его работы нуждается в дополнительной проверке и фильтрации.

Авторами с помощью пакета Bullet Physics Library был проведен расчет граничных точек области пересечения (находящихся на границе области пересечения) поверхностей пальца кисти Schunk SDH и объектов при захвате ОМ в виде детской игрушки и яблока.

Результаты расчета приведены на рис. 4 (точки обозначены зеленым). Из рис. 4, а видно наличие ложных граничных точек области пересечения, расположенных вне поверхностей моделей ЗУМ и ОМ; рис. 4, б иллюстрирует случай неполного определения граничных точек. Результаты моделирования, выполненного с использованием Open Dynamics Engine, аналогичны и обладают такими же недостатками.

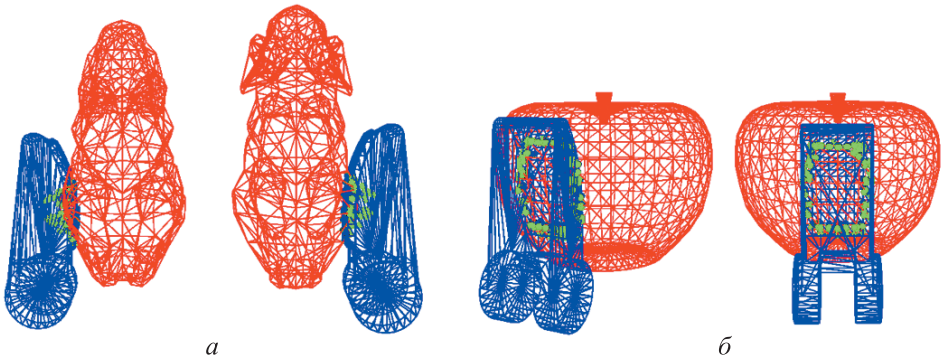


Рис. 4. Моделирование пересечения полигональных моделей звена ЗУ и ОМ с использованием Bullet Physics Library

Недостатком рассматриваемых алгоритмов является их применимость только к выпуклым полигональным моделям, что требует дополнительных вычислительных ресурсов для представления полигональной модели в виде некоторой декомпозиции, состоящей из выпуклых многогранников. Далее предлагается алгоритм решения задач узкой фазы, свободный от указанных недостатков.

Описание оригинального алгоритма узкой фазы.

Рассмотрим случай пересечения поверхностей двух объектов. Обозначим эти объекты соответственно А и В. Будем считать, что объект В перемещается относительно объекта А в направлении вектора v (рис. 5). Вопрос о выборе вектора v будет рассмотрен далее.

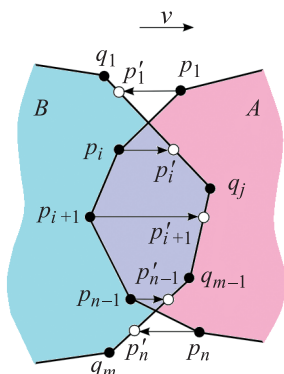


Рис. 5. Проекция вершин полигональной модели тела А на поверхность полигональной модели тела В вдоль вектора v

Пусть $P = \{p_1, \dots, p_n\}$ — множество вершин полигональной модели тела А. Пусть также $P' = \{p'_1, \dots, p'_n\}$ — соответствующее множество проекций вершин P на по-

верхность тела B . Проецирование осуществляется в направлении вектора v (см. рис. 5). Тогда, если вектор $p_i p'_i$, связывающий соответствующие элементы из множеств P и P' , сонаправлен с вектором v , то вершина p_i тела A находится в контакте с телом B , т. е. является граничной точкой области пересечения. Глубина проникновения для нее в направлении вектора v равна модулю вектора $p_i p'_i$, а вектор, вдоль которого отсчитывается глубина проникновения, сонаправлен с $p_i p'_i$.

Для нахождения проекции p'_i , соответствующей вершине p_i , предлагается из p_i провести прямую, параллельную вектору v , и определить из множества полигонов тела B тот, который эта прямая пересекает. Координаты точки пересечения будут соответствовать проекции p'_i .

При решении задачи поиска полигона, на котором будет располагаться проекция p_i , примем во внимание тот факт, что одна вершина (обозначим ее q_j) из множества вершин полигональной модели $Q = \{q_1, \dots, q_m\}$ тела B обладает таким свойством, что угол между векторами $p_i q_j$ и v будет наименьшим среди углов между p_i и любым другим вектором $q_k \in Q$. Данное свойство вытекает из того, что чем меньше угол $p'_i p_i q_j$, тем меньше расстояние между p'_i и q_j . Тогда, отыскав эту пару вершин p_i и q_j , можно избежать перебора всех полигонов тела B в поисках проекции p'_i и осуществлять поиск только среди полигонов, в которых q_j является одной из вершин, что существенно сократит время расчетов.

Таким образом, необходимо решить задачу поиска ближайшего соседа, которая заключается в отыскании среди множества элементов, расположенных в некотором метрическом пространстве, элементов, близких к заданному, согласно некоторой функции близости, определяющей это метрическое пространство. Полигональная модель в этом случае может рассматриваться как замкнутое метрическое пространство.

Среди алгоритмов поиска ближайших соседей можно выделить алгоритм разбиения пространства с помощью диаграммы Вороного [10], поиск в kD-деревьях [11], поиск в BSP-деревьях [12], поиск в VP-деревьях [13]. Сравнение перечисленных методов по таким параметрам, как вычислительная сложность, скорость поиска, сложность реализации, применимость в многомерном пространстве и адаптированность к замкнутым пространствам, показало, что наиболее применимым к решению задачи поиска ближайшего соседа в узкой фазе определения столкновения является алгоритм поиска в VP-деревьях.

Временная сложность алгоритма узкой фазы при использовании алгоритма поиска по VP-дереву вместо алгоритма Lin — Canny или V — Clip одинакова. Однако реализация алгоритма поиска по VP-дереву значительно проще. Кроме того, введение ОПВ сокращает время моделирования, так как в узкой фазе расчет проводится только для вершин, находящихся в ОПВ, а не для всех вершин полигональной модели.

Реализация алгоритма обнаружения пересечения объектов. На основе предложенного алгоритма разработано программное обеспечение, с помощью

которого реализован захват объекта трехпалой кистью Schunk SDH, имеющей семь степеней свободы и десять звеньев. Модель схвата создана на базе ROS в формате Unified Robot Description Format (URDF), который содержит информацию о кинематической цепи захватного устройства (ЗУ), типах шарниров, ориентации осей шарниров и длинах звеньев.

Решив прямую позиционную задачу, подробно рассмотренную в [14], определим положение и ориентацию звеньев кисти в инерциальной СК.

Полагаем, что геометрические поверхности ОМ звеньев ЗУ заданы с помощью полигональных моделей. Полигоны имеют форму треугольников, координаты вершин полигонов заданы в СК, совпадающей с СК объекта.

Алгоритм построения ОВВ. Моделирование захвата деформируемого объекта начинается с построения ОВВ звеньев ЗУМ и ОМ. Задача построения ОВВ для полигональной модели, заданной набором точек $P = \{p_1, \dots, p_n\}$, сводится к нахождению матрицы поворота $\tau_{obb\ obj}$ СК объекта к СК параллелепипеда, заданных в СК объекта координат $l_{obb\ obj}^{(obj)}$ центра параллелепипеда, и трех векторов $x_{obb}^{(obj)}$, $y_{obb}^{(obj)}$, $z_{obb}^{(obj)}$, определяющих направления и длины ребер параллелепипеда.

Сначала строится матрица M ковариации точек P , собственные векторы которой используются в качестве направляющих для ребер:

$$M = \frac{1}{n} \sum_{i=1}^n (p_i - \bar{p})(p_i - \bar{p})^T, \quad \bar{p} = \frac{1}{n} \sum_{i=1}^n p_i.$$

Собственные векторы матрицы M соответствуют ортам векторов $x_{obb}^{(obj)}$, $y_{obb}^{(obj)}$, $z_{obb}^{(obj)}$. Они находятся путем решения характеристического уравнения вида

$$(M - \lambda \cdot E)\alpha = 0,$$

где λ — собственные числа матрицы M ; E — единичная матрица; α — собственные векторы матрицы M , заданные в системе координат объекта.

Зная орты векторов, соответствующие направлению ребер ОВВ, нетрудно получить матрицу поворота $\tau_{obb\ obj}$ из СК объекта в СК соответствующего ему ОВВ по формуле:

$$\tau_{obb\ obj} = \begin{bmatrix} \frac{x_{obb}^{(obj)} x}{\|x_{obb}^{(obj)}\|} & \frac{x_{obb}^{(obj)} y}{\|x_{obb}^{(obj)}\|} & \frac{x_{obb}^{(obj)} z}{\|x_{obb}^{(obj)}\|} \\ \frac{y_{obb}^{(obj)} x}{\|y_{obb}^{(obj)}\|} & \frac{y_{obb}^{(obj)} y}{\|y_{obb}^{(obj)}\|} & \frac{y_{obb}^{(obj)} z}{\|y_{obb}^{(obj)}\|} \\ \frac{z_{obb}^{(obj)} x}{\|z_{obb}^{(obj)}\|} & \frac{z_{obb}^{(obj)} y}{\|z_{obb}^{(obj)}\|} & \frac{z_{obb}^{(obj)} z}{\|z_{obb}^{(obj)}\|} \end{bmatrix}.$$

Если полигональная модель выпуклая и расстояния между ее соседними вершинами приблизительно равны, то координаты центра ОВВ $l_{obb\ obj}^{(obj)}$ совпада-

ют со средним арифметическим вершин \bar{p} . Если полигональная модель невыпуклая, а расстояния между ее соседними вершинами значительно разнятся, то координаты $l_{obb\ obj}^{(obj)}$ находятся из соотношений

$$l_{obb\ obj\ x}^{(obj)} = \bar{p}_x + \frac{\max_{p_i \in P} \{ [p_i - \bar{p}]_x \} + \min_{p_i \in P} \{ [p_i - \bar{p}]_x \}}{2},$$

$$l_{obb\ obj\ y}^{(obj)} = \bar{p}_y + \frac{\max_{p_i \in P} \{ [p_i - \bar{p}]_y \} + \min_{p_i \in P} \{ [p_i - \bar{p}]_y \}}{2},$$

$$l_{obb\ obj\ z}^{(obj)} = \bar{p}_z + \frac{\max_{p_i \in P} \{ [p_i - \bar{p}]_z \} + \min_{p_i \in P} \{ [p_i - \bar{p}]_z \}}{2}.$$

Длины векторов $x_{obb}^{(obj)}$, $y_{obb}^{(obj)}$, $z_{obb}^{(obj)}$ определяются отысканием множества максимально удаленных от центра ОБВ точек P по направлениям, соответствующим этим векторам

$$\|x_{obb}^{(obj)}\| = \max_{p_i \in P} \left\{ \left[\tau_{obb\ obj} \cdot (p_i - l_{obb\ obj}^{(obj)}) \right]_x \right\},$$

$$\|y_{obb}^{(obj)}\| = \max_{p_i \in P} \left\{ \left[\tau_{obb\ obj} \cdot (p_i - l_{obb\ obj}^{(obj)}) \right]_y \right\},$$

$$\|z_{obb}^{(obj)}\| = \max_{p_i \in P} \left\{ \left[\tau_{obb\ obj} \cdot (p_i - l_{obb\ obj}^{(obj)}) \right]_z \right\}.$$

Координаты вершин $a_{iobb\ obb}^{(obj)}$ ОБВ в СК объекта будут иметь вид:

$$a_{1obb\ obb}^{(obj)} = l_{obb\ obj}^{(obj)} - x_{obb}^{(obj)} - y_{obb}^{(obj)} - z_{obb}^{(obj)}, \quad a_{2obb\ obb}^{(obj)} = l_{obb\ obj}^{(obj)} + x_{obb}^{(obj)} - y_{obb}^{(obj)} - z_{obb}^{(obj)},$$

$$a_{3obb\ obb}^{(obj)} = l_{obb\ obj}^{(obj)} + x_{obb}^{(obj)} + y_{obb}^{(obj)} - z_{obb}^{(obj)}, \quad a_{4obb\ obb}^{(obj)} = l_{obb\ obj}^{(obj)} - x_{obb}^{(obj)} + y_{obb}^{(obj)} - z_{obb}^{(obj)},$$

$$a_{5obb\ obb}^{(obj)} = l_{obb\ obj}^{(obj)} - x_{obb}^{(obj)} - y_{obb}^{(obj)} + z_{obb}^{(obj)}, \quad a_{6obb\ obb}^{(obj)} = l_{obb\ obj}^{(obj)} + x_{obb}^{(obj)} - y_{obb}^{(obj)} + z_{obb}^{(obj)},$$

$$a_{7obb\ obb}^{(obj)} = l_{obb\ obj}^{(obj)} + x_{obb}^{(obj)} + y_{obb}^{(obj)} + z_{obb}^{(obj)}, \quad a_{8obb\ obb}^{(obj)} = l_{obb\ obj}^{(obj)} - x_{obb}^{(obj)} + y_{obb}^{(obj)} + z_{obb}^{(obj)}.$$

Результат построения ОБВ для ОМ и звеньев ЗУМ представлен на рис. 6.

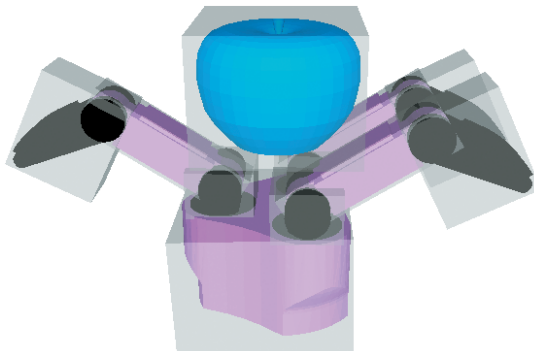


Рис. 6. Ориентированные ОП, построенные для звеньев ЗУМ и ОМ

Алгоритм вычисления векторов направления относительного движения.

Этот алгоритм определяется особенностями конструкции ЗУМ — векторы соответствуют перемещениям, которые могут совершать звенья захватного устройства при захвате объектов.

Для каждого звена ЗУМ рассчитывается вектор направления движения, заданный в СК звена. С учетом того, что между звеном k и базовым звеном находятся другие звенья кинематической цепи, результирующий вектор $v_{\Sigma k}^{(33Yk)}$ рассчитывается из соотношения:

$$v_{\Sigma k}^{(33Yk)} = \sum_{i=1}^k e_i b_i \tau_{k0} \tau_{i0}^T v_i^{(33Yi)} = \sum_{i=1}^k e_i b_i \tau_{ki} v_i^{(33Yi)},$$

где e_i — индикатор наличия движения шарнира, приводящего в движение звено i ($e_i = 1$, если шарнир, к которому прикреплено звено i , изменил свое положение на данном шаге моделирования, и $e_i = 0$, если приводящий в движение звено i шарнир остался неподвижен); b_i — скалярная величина, равная модулю линейной скорости i -го звена, если шарнир, к которому прикреплено звено i , поступательного типа, или модулю угловой скорости, если шарнир вращательного типа; τ_{i0} и τ_{k0} — транспонированные матрицы поворота СК i -го и k -го звеньев; τ_{ki} — матрица поворота СК звена i относительно СК звена k ; $v_i^{(33Yi)}$ — вектор направления движения i -го звена относительно предыдущего звена кинематической цепи (звена, к которому прикреплен шарнир, приводящий в движение звено i).

Возможны три варианта вектора $v_i^{(33Yi)}$. Если шарнир, к которому прикреплено звено, поступательного типа, то вектор $v_i^{(33Yi)}$ совпадает по направлению с осью шарнира и $\|v_i^{(33Yi)}\| = 1$. Если шарнир, к которому прикреплено звено, вращательного типа, то $v_i^{(33Yi)}$ есть векторное произведение вектора оси шарнира и вектора длины звена $l_{i+1i}^{(i)}$, который соединяет СК звена i и СК следующего звена кинематической цепи (в случае, когда угол между $l_{i+1i}^{(i)}$ и осью шарнира является малым, а значит $\|v_i^{(33Yi)}\| \rightarrow 0$, вектор направления движения принимается нулевым). Если звено не имеет шарнира (базовое звено кинематической цепи), то $v_i^{(33Yi)}$ — нулевой.

Векторы $v_i^{(33Yi)}$ определяются в начале моделирования, а векторы $v_{\Sigma k}^{(33Yk)}$ рассчитываются на каждом шаге моделирования захвата.

В узкой фазе для отыскания проекций вершин полигональной модели звена k ЗУМ на полигональную модель ОМ используется вектор направления относительного движения, равный $-v_{\Sigma k}^{(33Y)}$, как если бы звено было неподвижно, а двигался бы объект. Для определения проекций вершин полигональной модели ОМ на полигональную модель k -го звена ЗУМ в качестве вектора относительно направления движения принимается вектор $v_k^{(OM)}$, заданный в СК ОМ:

$$v_k^{(OM)} = \tau_{OM0} \tau_{k0}^T v_{\Sigma k}^{(33Yk)} = \tau_{OMk} v_{\Sigma k}^{(33Yk)},$$

где τ_{OM0} и τ_{k0} — матрицы поворота из инерциальной СК в СК ОМ и k -го звена соответственно; τ_{OMk} — матрицы поворота в СК ОМ из СК звена k .

Далее для простоты изложения вектор относительного перемещения будем обозначать как v .

Широкая фаза алгоритма обнаружения пересечения объектов. Пусть положение ОМ и звена ЗУМ на текущем шаге моделирования задано в инерциальной СК с помощью векторов линейных смещений $l_{OM0}^{(0)}$, $l_{33Y0}^{(0)}$ и матриц поворота τ_{OM0} , τ_{33Y0} соответственно. Положения параллелепипедов ОБВ относительно СК объектов (ОМ и звена ЗУМ) заданы с помощью векторов линейных смещений $l_{obbOM\ OM}^{(OM)}$, $l_{obb33Y\ 33Y}^{(33Y)}$ и матриц поворота относительно СК объектов $\tau_{obbOM\ OM}$, $\tau_{obb33Y\ 33Y}$ соответственно.

Для определения областей пересечения двух параллелепипедов необходимо координаты их вершин перевести из СК объектов в СК одного из ОБВ. Перевод из СК объектов в СК ОП ОМ осуществляется по формуле

$$a_{iobb33Y\ OM}^{(obbOM)} = \tau_{obbOM\ OM} \left(\tau_{OM0}^T \left(\tau_{33Y0} a_{iobb33Y\ 33Y}^{(obb33Y)} - l_{OM0}^{(0)} + l_{33Y0}^{(0)} \right) \right),$$

$$a_{iobbOM\ OM}^{(obbOM)} = \tau_{obbOM\ OM} a_{iobbOM\ OM}^{(OM)}, \quad i=1, \dots, 8.$$

Перевод из СК объектов в СК ОП звена ЗУМ:

$$a_{iobbOM\ 33Y}^{(obb33Y)} = \tau_{obb33Y\ 33Y} \left(\tau_{33Y0}^T \left(\tau_{OM0} a_{iobbOM\ OM}^{(obbOM)} - l_{33Y0}^{(0)} + l_{OM0}^{(0)} \right) \right),$$

$$a_{iobb33Y\ 33Y}^{(obb33Y)} = \tau_{obb33Y\ 33Y} a_{iobb33Y\ 33Y}^{(33Y)}, \quad i=1, \dots, 8.$$

При этом если выполняются неравенства для случая перевода в СК ОП ОМ:

$$\min \{ a_{iobbOM\ OM_x}^{(obbOM)} \} > \max \{ a_{iobb33Y\ OM_x}^{(obbOM)} \}, \quad \max \{ a_{iobbOM\ OM_x}^{(obbOM)} \} < \min \{ a_{iobb33Y\ OM_x}^{(obbOM)} \},$$

$$\min \{ a_{iobbOM\ OM_y}^{(obbOM)} \} > \max \{ a_{iobb33Y\ OM_y}^{(obbOM)} \}, \quad \max \{ a_{iobbOM\ OM_y}^{(obbOM)} \} < \min \{ a_{iobb33Y\ OM_y}^{(obbOM)} \},$$

$$\max \{ a_{iobbOM\ OM_z}^{(obbOM)} \} > \max \{ a_{iobb33Y\ OM_z}^{(obbOM)} \}, \quad \max \{ a_{iobbOM\ OM_z}^{(obbOM)} \} < \min \{ a_{iobb33Y\ OM_z}^{(obbOM)} \}$$

и для случая перевода в СК ОП звена ЗУМ

$$\min \{ a_{iobb33Y\ 33Y_x}^{(obb33Y)} \} > \max \{ a_{iobbOM\ 33Y_x}^{(obb33Y)} \}, \quad \max \{ a_{iobb33Y\ 33Y_x}^{(obb33Y)} \} < \min \{ a_{iobbOM\ 33Y_x}^{(obb33Y)} \},$$

$$\min \{ a_{iobb33Y\ 33Y_y}^{(obb33Y)} \} > \max \{ a_{iobbOM\ 33Y_y}^{(obb33Y)} \}, \quad \max \{ a_{iobb33Y\ 33Y_y}^{(obb33Y)} \} < \min \{ a_{iobbOM\ 33Y_y}^{(obb33Y)} \},$$

$$\min \{ a_{iobb33Y\ 33Y_z}^{(obb33Y)} \} > \max \{ a_{iobbOM\ 33Y_z}^{(obb33Y)} \}, \quad \max \{ a_{iobb33Y\ 33Y_z}^{(obb33Y)} \} < \min \{ a_{iobbOM\ 33Y_z}^{(obb33Y)} \},$$

то пересечения нет, устанавливается факт отсутствия контакта между ОМ и звеном ЗУМ на текущем шаге моделирования.

Если одно или несколько условий не выполняются, то в ОП выделяется область потенциального взаимодействия. Она формируется границами ОБВ, в СК которого осуществлялся перевод, а также плоскостями, заданными координатами вершин второго ОБВ, при проверке которых условия не выполнялись, и нормальными, соответствующими ортам осей СК первого ОБВ, по которым условия не были выполнены (см. рис. 3, а).

Затем устанавливается факт наличия в ОПВ вершин полигональной модели соответствующего объекта. Если ни одна из вершин не располагается внутри ОПВ, делается вывод, что на текущем шаге моделирования контакт между ОМ и звеном ЗУ отсутствует. В этом случае вычисления продолжаются в широкой фазе.

Если же в ОПВ обоих ОП находятся вершины полигональных моделей обоих объектов $\tilde{P} = \{\tilde{p}_1, \dots, \tilde{p}_f\}$, $\tilde{P} \subseteq P$ и $\tilde{Q} = \{\tilde{q}_1, \dots, \tilde{q}_g\}$, $\tilde{Q} \subseteq Q$ (см. рис. 3, б), в рамках широкой фазы делается вывод о возможном наличии пересечения, и для этого шага моделирования выполняется узкая фаза определения пересечения, которая выявляет факт наличия или отсутствия пересечения, а также при выявлении пересечения рассчитывает точки, лежащие на поверхности пересечения, соответствующие им глубины и нормали проникания.

Узкая фаза алгоритма обнаружения пересечения объектов. Использование расстояния между точками \tilde{p}_i и \tilde{q}_j в качестве функции близости вершин проще в реализации и понимании, чем использование угла между векторами $\tilde{p}_i\tilde{q}_j$ и v . Очевидно, что угол между векторами $\tilde{p}_i\tilde{q}_j$ и v будет наименьшим, если проекции вершин \tilde{p}_i и \tilde{q}_j на плоскость, заданную вектором v в качестве вектора нормали и проходящую через точку \tilde{p}_i вдоль направления относительного движения v , будут ближайшими соседями с точки зрения расстояния между ними. Проецирование множеств точек \tilde{P} и \tilde{Q} дает два множества точек — $\tilde{P}' = \{\tilde{p}'_1, \dots, \tilde{p}'_f\}$ и $\tilde{Q}' = \{\tilde{q}'_1, \dots, \tilde{q}'_g\}$, элементы которых удовлетворяют равенствам:

$$\tilde{p}'_i = \tilde{p}_i - v \frac{v_x (\tilde{p}_{ix} - \tilde{p}_{1x}) + v_y (\tilde{p}_{iy} - \tilde{p}_{1y}) + v_z (\tilde{p}_{iz} - \tilde{p}_{1z})}{v_x^2 + v_y^2 + v_z^2}, \quad i=1, \dots, f,$$

$$\tilde{q}'_j = \tilde{q}_j - v \frac{v_x (\tilde{q}_{jx} - \tilde{p}_{1x}) + v_y (\tilde{q}_{jy} - \tilde{p}_{1y}) + v_z (\tilde{q}_{jz} - \tilde{p}_{1z})}{v_x^2 + v_y^2 + v_z^2}, \quad j=1, \dots, g.$$

Для решения задачи поиска ближайшего соседа для множеств \tilde{P}' и \tilde{Q}' строятся VP-деревья. Построение VP-дерева является рекурсивной процедурой — на каждой итерации алгоритма статистически [13] выбирается опорная точка и вычисляется среднее расстояние от нее до всех остальных точек — радиус области узла R . Затем остальные точки делятся на два подмножества, называемых

поддеревьями. Если расстояние между вершиной и опорной точкой меньше R , то она принадлежит внутреннему поддереву, иначе — внешнему. В каждом из получившихся подмножеств выбирается следующая опорная точка и рассчитывается новый радиус. Операция построения дерева продолжается до тех пор, пока размеры подмножеств не достигнут заданного предела — максимального размера листа [13].

Затем путем поиска по VP-деревьям для каждого элемента множеств \tilde{P}' и \tilde{Q}' определяется ближайший сосед среди элементов множеств \tilde{Q}' и \tilde{P}' . Поиск по VP-дереву осуществляется следующим образом — начиная с корневого узла дерева определяется расстояние между вершиной, для которой осуществляется поиск, и центром узла VP-дерева. Если расстояние меньше радиуса области узла, то поиск продолжается во внутреннем поддереве, иначе — во внешнем. По достижении листа дерева выполняется линейный поиск среди точек листа [13].

Далее, опираясь на индексы полученных пар ближайших соседей из множеств \tilde{P}' и \tilde{Q}' , восстанавливаются индексы для соответствующих точек из \tilde{P} и \tilde{Q} и в итоге для P и Q . На этом этапе проводится дополнительная фильтрация — если угол между векторами $p_i q_j$ и v — тупой, то очевидно, что точка p_i тела A не находится в контакте с телом B , и ее можно исключить из дальнейшего рассмотрения (вершины p_1 и q_1 на рис. 5).

Для конфигурации ЗУМ, изображенной на рис. 7, а, представлен результат поиска ближайшего соседа по VP-дереву для вершин полигональных моделей, попавших в ОПВ. На рис. 7, а зелеными линиями изображен ОПВ для ОП ОМ, красными — ОПВ для ОП звена ЗУМ. На рис. 7, б синими точками обозначены вершины ОМ, находящиеся внутри ОПВ, соответствующей ОБВ ОМ, розовыми — вершины звена ЗУМ, расположенные в ОПВ ОБВ звена ЗУ, синими и розовыми линиями показаны соответственно ближайшие соседи для вершин ОМ и звена ЗУ после фильтрации.

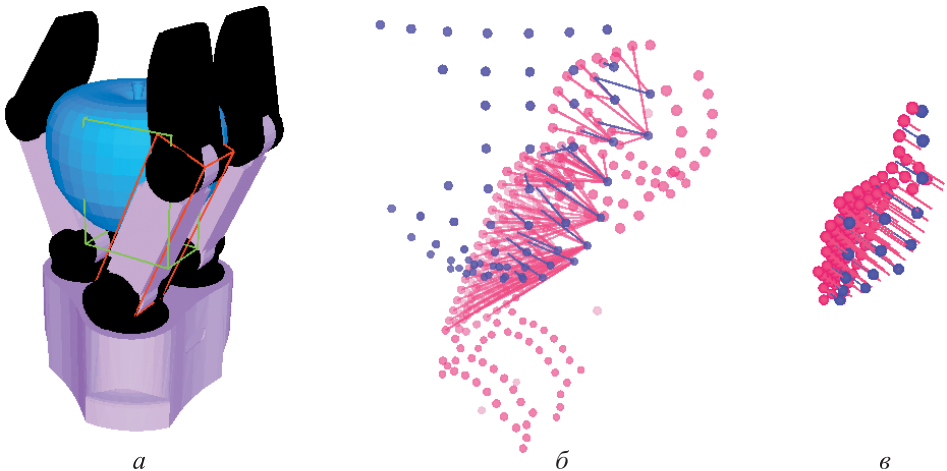


Рис. 7. Этапы узкой фазы определения пересечения ОМ и звена ЗУ

Затем через p_i строится прямая, параллельная вектору направления относительного движения v . Далее среди граней второго объекта отыскиваются полигоны, одной из вершин которых является q_j . Для каждого из этих полигонов рассчитывается точка пересечения построенной прямой с плоскостью, совпадающей с гранью, и устанавливается принадлежность полученной точки полигону.

Точка пересечения прямой с плоскостью p'_i , проходящей через k -й полигон с нормалью n_k и вершинами q_j, q_{j1}, q_{j2} , находится с помощью соотношения

$$p'_i = p_i - v \frac{n_{kx}(p_{ix} - q_{jx}) + n_{ky}(p_{iy} - q_{jy}) + n_{kz}(p_{iz} - q_{jz})}{n_{kx}v_x + n_{ky}v_y + n_{kz}v_z},$$

$$n_k = \frac{(q_{j1} - q_j) \times (q_{j2} - q_j)}{\|(q_{j1} - q_j) \times (q_{j2} - q_j)\|}.$$

Точка p'_i будет принадлежать полигону k , когда площадь треугольника $\Delta q_j q_{j1} q_{j2}$ будет равна площадям треугольников $\Delta p'_i q_j q_{j1}$, $\Delta p'_i q_{j1} q_{j2}$ и $\Delta p'_i q_{j2} q_j$:

$$S_{\Delta q_j q_{j1} q_{j2}} = S_{\Delta p'_i q_j q_{j1}} + S_{\Delta p'_i q_{j1} q_{j2}} + S_{\Delta p'_i q_{j2} q_j}.$$

Если вершина p'_i принадлежит этому полигону, то она является проекцией точки p_i на поверхность полигональной модели второго объекта B вдоль вектора направления относительного движения v второго тела относительно первого. Если вектор $p_i p'_i$ сонаправлен с вектором v , то вершина p_i лежит на границе области пересечения. Соответствующий точке p_i вектор, вдоль которого отсчитывается глубина проникновения, совпадает по направлению с вектором $p_i p'_i$, а глубина проникновения равна модулю вектора $p_i p'_i$ (рис. 7, в).

Проделав эти вычисления для каждой из вершин, расположенной в ОПВ, определяем точки, лежащие на границе области пересечения, соответствующие им глубины и нормали проникновения. Результат работы предложенного алгоритма для всех звеньев ЗУМ представлен на рис. 8. Справа на рисунке синими и

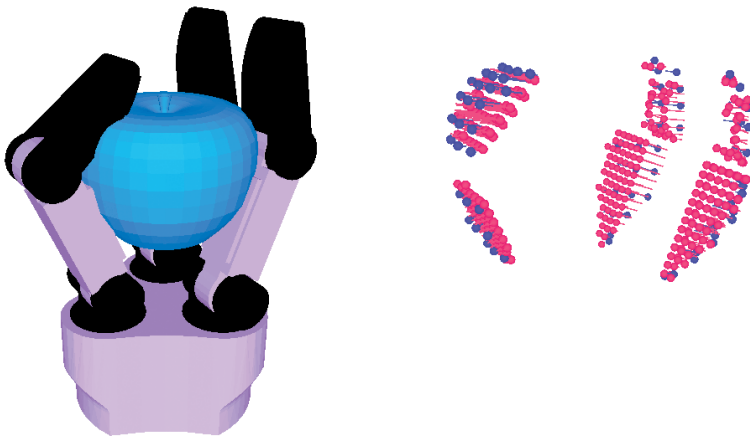


Рис. 8. Результат работы предложенного алгоритма

розовыми точками обозначены вершины ОМ и звеньев ЗУМ, находящиеся внутри пятен контакта, синие и розовые линии соответствуют векторам $p_i p_i'$ и $q_i q_i'$.

Заключение. Таким образом, результаты моделирования подтвердили правильность и эффективность предложенного алгоритма.

Введение областей потенциального взаимодействия повышает точность широкой фазы определения пересечения вследствие проверки наличия вершин полигональных моделей внутри соответствующих ОПВ, поскольку в процессе моделирования уменьшается число случаев, когда имеет место пересечение ОП, но пересечение между полигональными моделями отсутствует.

Показано, что использование в узкой фазе алгоритма поиска по VP-дереву вместо V-Clip или Lin-Canny незначительно повлияло на временную сложность метода. Но поскольку время вычислений уменьшается вместе с количеством точек, для которых происходят вычисления, введение ОПВ сокращает время моделирования, так как в узкой фазе расчет проводится только для вершин, находящихся в ОПВ, а не для всех вершин полигональной модели.

Существенным достоинством предложенного алгоритма является его применимость к невыпуклым полигональным моделям без необходимости их представления в виде некоторой декомпозиции, состоящей из выпуклых многогранников.

Форма области пересечения, глубины и нормали проникания могут быть использованы для определения геометрии пятен контакта и эпюр нагрузки, которые участвуют в расчете сил контактного взаимодействия между ЗУМ и ОМ. По известным силам контактного взаимодействия можно оценить качество захвата и предсказать поведение ОМ в процессе выполнения операции.

ЛИТЕРАТУРА

1. *Аппаратно-программный комплекс для решения задач автоматического захвата объекта манипуляторами* / А.Г. Лесков, В.В. Илларионов, И.А. Калеватых, С.Д. Мо-рошкин, К.В. Бажинова, Е.В. Феоктистова // Инженерный журнал: наука и инновации. 2015. Вып. 1. DOI: 10.18698/2308-6033-2015-1-1361 URL: <http://engjournal.ru/catalog/pribor/robot/1361.html>
2. *Boivin E., Sharf I., Doyon M.* Optimum grasp of planar and revolute objects with gripper geometry constraints // Proc. IEEE ICRA 2004. 2004. P. 326–332. DOI: 10.1109/ROBOT.2004.1307171
3. *Pauly M., Pai D.K., Guibas L.J.* Quasi-Rigid Objects in Contact // Eurographics/ACM SIGGRAPH Symposium on Computer Animation. 2004. P. 109–119.
4. *Ericson C.* Real-time collision detection. San Francisco: Elsevier, 2005. 593 p.
5. *Gilbert E.G., Johnson D.W., Keerthi S.S.* A fast procedure for computing the distance between complex objects in three-dimensional space // IEEE Journal of Robotics and Automation. 1988. Vol. 4. No. 2. P. 193–203. DOI: 10.1109/56.2083
6. *Lin M.C.* Efficient collision detection for animation and robotics. PhD dissertation. University of California, Berkeley, CA, USA, 1993. 159 p.
7. *Mirtich B.* V-Clip: fast and robust polyhedral collision detection // Journal ACM Transactions on Graphics. 1998. Vol. 17. No. 3. P. 177–208.

8. *Bullet Physics Library*. URL: <http://bulletphysics.org/> (дата обращения 25.12.2015).
9. *Open Dynamics Engine*. URL: <http://www.ode.org/> (дата обращения 25.12.2015).
10. *Пренарата Ф., Шеймос М.* Вычислительная геометрия. М.: Мир, 1989. 478 с.
11. *Panigrahy R.* An improved algorithm finding nearest neighbor using Kd-trees // Proc. 8th Latin American Symposium. Búzios, Brazil, 2008. Vol. 26. No. 4. P. 387–398.
12. *Maneewongvatana S., Mount D.M.* An empirical study of a new approach to nearest neighbor searching // Proc. 3th International workshop on algorithm engineering and experimentation. Washington, DC, USA, 2001. P. 172–187.
13. *Yianilos P.N.* Data structures and algorithms for nearest neighbor search in general metric spaces // Proc. 4th annual ACM-SIAM symposium on discrete algorithms. Austin, TX, USA, 1993. P. 311–321.
14. *Лесков А.Г., Бажинова К.В., Морошкин С.Д., Феоктистова Е.В.* Построение моделей кинематики исполнительных механизмов манипуляционных роботов с использованием блочных матриц // Инженерный журнал: наука и инновации. 2013. Вып. 9. DOI: 10.18698/2308-6033-2013-9-954 URL: <http://engjournal.ru/catalog/pribor/robot/954.html>

Лесков Алексей Григорьевич — д-р техн. наук, профессор, директор Дмитровского филиала МГТУ им. Н.Э. Баумана (Российская Федерация, 141801, Московская обл., Дмитровский р-н, пос. Орево).

Селиверстова Елена Владимировна — инженер Дмитровского филиала МГТУ им. Н.Э. Баумана (Российская Федерация, 141801, Московская обл., Дмитровский р-н, пос. Орево), аспирантка кафедры «Робототехнические системы и мехатроника» МГТУ им. Н.Э. Баумана (Российская Федерация, 105005, Москва, 2-я Бауманская ул., д. 5).

Просьба ссылаться на эту статью следующим образом:

Лесков А.Г., Селиверстова Е.В. Расчет областей пересечения поверхностей захватных устройств манипуляторов и деформируемых объектов при планировании и моделировании захвата // Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение. 2016. № 6. С. 97–114. DOI: 10.18698/0236-3933-2016-6-97-114

COMPUTATION OF INTERSECTION REGIONS OF ROBOTIC GRIPPER AND DEFORMABLE OBJECT SURFACES DURING GRASP PLANNING AND SIMULATION

A.G. Leskov¹

agleskov@rambler.ru

E.V. Seliverstova^{1,2}

feoktistovaev@mail.ru

¹ Dmitrov Branch of Bauman Moscow State Technical University, Orevo, Moscow Region, Russian Federation

² Bauman Moscow State Technical University, Moscow, Russian Federation

Abstract

The study tested methods for detecting and computing the intersection regions of robotic gripper and deformable object surfaces during the grasp planning and simulation. We suggest an algorithm for detection and computation of

Keywords

Grip, deformable object, simulating, intersection of polygonal models, Oriented Bounding Box, VP-tree

the intersection regions comprising both broad and narrow phases. The broad phase algorithm is based on the bounding box method and is improved by introducing the original algorithm for detecting areas of potential interaction between the gripper and the object. The narrow phase algorithm is novel. It uses the nearest neighbor search methods and considers the movement direction of interacting bodies. As a result of our research, we developed a computer program and carried out experiments to analyze the efficiency of the proposed algorithms for the grasp planning with the 3-finger robotic hand Schunk SDH

REFERENCES

- [1] Leskov A.G., Illarionov V.V., Kalevatykh I.A., Moroshkin S.D., Bazhinova K.V., Feoktistova E.V. Hardware-software complex for solving the task of automatic capture of the object with manipulators. *Inzhenernyy zhurnal: nauka i innovatsii* [Engineering Journal: Science and Innovation], 2015, iss. 1 (in Russ.). DOI: 10.18698/2308-6033-2015-1-1361
Available at: <http://engjournal.ru/eng/catalog/pribor/robot/1361.html>
- [2] Boivin E., Sharf I., Doyon M. Optimum grasp of planar and revolute objects with gripper geometry constraints. *Proc. ICRA 2004*, 2004, pp. 326–332.
DOI: 10.1109/ROBOT.2004.1307171
- [3] Pauly M., Pai D.K., Guibas L.J. Quasi-rigid objects in contact. *Proc. 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 2004, pp. 109–119.
- [4] Ericson C. Real-time collision detection. San Francisco, Elsevier, 2005. 593 p.
- [5] Gilbert E.G., Johnson D.W., Keerthi S.S. A fast procedure for computing the distance between complex objects in three-dimensional space. *IEEE Journal of Robotics and Automation*, 1988, vol. 4, no. 2, pp. 193–203. DOI: 10.1109/56.2083
- [6] Lin M.C. Efficient collision detection for animation and robotics. PhD dissertation. University of California, Berkeley, CA, USA, 1993. 159 p.
- [7] Mirtich B. V-Clip: fast and robust polyhedral collision detection. *Journal ACM Transactions on Graphics*, 1998, vol. 17, no. 3, pp. 177–208.
- [8] Bullet Physics Library. Available at: <http://bulletphysics.org/> (accessed 25.12.2015)
- [9] Open Dynamics Engine. Available at: <http://www.ode.org/> (accessed 25.12.2015)
- [10] Preparata F.P., Shamos M.I. Computational geometry: an introduction. New York, Spinger-Verlag. 412 p. (Russ. ed.: Vychislitelnaya geometriya. Moscow, Mir Publ., 1989. 478 p.)
- [11] Panigrahy R. An improved algorithm finding nearest neighbor using Kd-trees. *Proc. 8th Latin American Symposium*, Búzios, Brazil, 2008, vol. 26, no. 4, pp. 387–398.
- [12] Maneewongvatana S., Mount D.M. An empirical study of a new approach to nearest neighbor searching. *Proc. third international workshop on algorithm engineering and experimentation*, Washington, DC, USA, 2001, pp. 172–187.
- [13] Yianilos P.N. Data structures and algorithms for nearest neighbor search in general metric spaces. *Proc. fourth annual ACM-SIAM Symposium on Discrete algorithms*, Austin, TX, USA, 1993, pp. 311–321.

[14] Leskov A.G., Bazhinova K.V., Moroshkin S.D., Feoktistova E.V. Modeling of trobotic arms kinematics by means of block matrixes. *Inzhenernyy zhurnal: nauka i innovatsii* [Engineering Journal: Science and Innovation], 2013, iss. 9, no. 21 (in Russ.).

DOI: 10.18698/2308-6033-2013-9-954 Available at: <http://engjournal.ru/eng/catalog/pribor/robot/954.html>

Leskov A.G. — Dr. Sci. (Eng.), Professor, Director of Dmitrov Branch of Bauman Moscow State Technical University (Orevo, Dmitrovsky District, Moscow Region, 141801 Russian Federation).

Seliverstova E.V. — engineer of Dmitrov Branch of Bauman Moscow State Technical University (Orevo, Dmitrovsky District, Moscow Region, 141801 Russian Federation), post-graduate student of Robotic Systems and Mechatronics Department, Bauman Moscow State Technical University (2-ya Baumanskaya ul. 5, Moscow, 105005 Russian Federation).

Please cite this article in English as:

Leskov A.G., Seliverstova E.V. Computation of Intersection Regions of Robotic Gripper and Deformable Object Surfaces During Grasp Planning and Simulation. *Vestn. Mosk. Gos. Tekh. Univ. im. N.E. Baumana, Priborostr.* [Herald of the Bauman Moscow State Tech. Univ., Instrum. Eng.], 2016, no. 6, pp. 97–114. DOI: 10.18698/0236-3933-2016-6-97-114



В Издательстве МГТУ им. Н.Э. Баумана
вышло в свет учебное пособие автора

В.В. Бондарев

**«Введение в информационную
безопасность автоматизированных систем»**

Рассмотрена законодательная база информационной безопасности, приведен перечень возможных угроз, отражены основные подходы к созданию систем защиты информации, представлена классификация предупредительных мер, изучены вопросы, связанные с программно-аппаратными механизмами обеспечения информационной безопасности.

По вопросам приобретения обращайтесь:

105005, Москва, 2-я Бауманская ул., д. 5, стр. 1

+7 (499) 263-60-45

press@bmstu.ru

www.baumanpress.ru