УДК 67.02, 004.942, 519.178

ФОРМАЛЬНАЯ ПОСТАНОВКА ЗАДАЧИ СБОРКИ СЛОЖНЫХ ИЗДЕЛИЙ

А.Н. Божко
Московский государственный технический университет им. Н.Э. Баумана
Россия, 105005, Москва, 2-я Бауманская ул., д. 5, стр. 1
E-mail: abozhko@inbox.ru

Ключевые слова: автоматизация проектирования, сборка, последовательность сборки, планирование перемещений, конфигурационное пространство, CAAP, CAD.

Аннотация: Работа посвящена автоматизации проектирования процессов сборки технических систем. Описана математическая постановка задачи сборки в современной парадигме Motion Planning. Движение устанавливаемой детали в служебное положение представлено как перемещение точки в многомерном конфигурационном пространстве состояний собираемого изделия. Предложены математические описания базовых свойств сборочного процесса: захват, устойчивость, когерентность, собираемость и др. Предложенная модель корректно описывает геометрические аспекты сборочного процесса и позволяет использовать глубоко развитые методы и алгоритмы планирования перемещений для синтеза траекторий перемещения деталей в процессе сборки изделия.

1. Введение

Техническая подготовка сборочного производства является концентратором связей между конструкторскими и технологическими стадиями жизненного цикла технических систем. Для изделий средней и высокой сложности процесс сборки разрабатывается до технологических процессов обработки деталей. В процессе синтеза сборочных операций и переходов происходит верификация конструкции и уточняются технические требования к процессам изготовления деталей. Поэтому автоматизация проектирования процесса сборки сложных технических систем – это важная и актуальная проблема современных информационных технологий.

В настоящее время автоматизация проектирования сборочных процессов – это динамично развивающийся раздел информационных технологий, который располагает значительным массивом публикаций и большим числом программных разработок. Для решения этой сложной проблемы применяют большое число моделей и методов из различных отраслей информатики и дискретной математики: теория графов, искусственный интеллект, программирование роботов, комбинаторная геометрия, математическая логика, общая и булева алгебра, теория баз данных, машинная графика, анализ столкновений (Collision detection), планирование перемещений (Motion planning) и др. [1–3].

В работе предлагается формальная постановка задачи сборки изделия, которое рассматривается как геометрическая система, состоящая из абсолютно твердых и невесомых элементов, соединенных разъёмными механическими связями (соединениями). Задача синтеза траекторий деталей в пространстве собираемого изделия может быть поставлена как частный случай более общей проблемы – планирование перемещений элементов геометрической системы в пространстве состояний (Motion planning, Path planning) [4].

2. Основные понятия и обозначения

Введем необходимые понятия и обозначения. Пусть в трехмерном пространстве *E* задано множество $Parts = \{p_1, p_2, \ldots, p_n\}$ подвижных объектов (деталей) и множество препятствий $Obs = \{o_1, o_2, \ldots, o_l\}$, положение которых фиксировано. Препятствия и объекты рассматриваем как абсолютно твердые тела. Пусть существует некоторый агент, который выполняет захват и перемещение объектов в пространстве *E*. Назовем его роботом и обозначим *Rbt*. Определим конфигурационное пространство (пространство состояний) геометрической системы $Sys = Parts \cup Obs \cup Rbt$, состоящей из объектов, препятствий и робота, следующим образом:

- свяжем с каждым объектом $p_i, i = \overline{1, n}$ из множества *Parts* и роботом *Rbt* собственную систему координат;
- обозначим *R* – *r*-мерное конфигурационное пространство робота *Rbt*, где *r* – число степеней свободы робота;
- обозначим P_i– 6-мерное конфигурационное пространство объекта $p_i, i = \overline{1, n}$;
- образуем декартово произведение $P = P_1 \times P_2 \times \ldots \times P_n$ конфигурационных пространств объектов. Пространство *P* представляет собой $(6 \times n)$-мерное пространство, точки которого соответствуют конфигурациям, образованным из элементов множества *Parts*;

Рассмотрим декартово произведение $C = R \times P$. C представляет собой конфигурационное пространство объединенной системы $Parts \cup Rbt$. Любая точка этого пространства $c \in C$ задает конфигурацию, которой соответствует определенные позиции объектов $p_i, i = \overline{1, n}$ и робота *Rbt* в физическом трехмерном пространстве *E*.

Обозначим $E_i(c)$ и $Rbt(c)$ – позиции, которые занимают объект p_i робот *Rbt* в пространстве *E* в конфигурации *c*, а $E_i(c)/E_j(c)$ – относительное положение объектов p_i и p_j в точке конфигурационного пространства *c*. Очевидно, что $E_i(c)/E_j(c) = E_j(c)/E_i(c)$ для любых i и j.

Все $E_i(c)$ и $Rbt(c)$ представляют собой трехмерные многообразия с границей в пространстве *E* [5]. Обозначим *int* (*M*) и *bound* (*M*) – множество внутренних и соответственно граничных точек многообразия *M*. Два объекта p_i и p_j пересекаются в конфигурации *c*, если $int(E_i(c)) \cap int(E_j(c)) \neq \emptyset$.

Два объекта p_i и p_j контактируют в конфигурации *c*, если $\left(int(E_i(c)) \cap int\left(E_j(c)\right) = \emptyset\right) \wedge \left(bound(E_i(c)) \cap bound\left(E_j(c)\right) \neq \emptyset\right)$. Таким же образом задаются отношения пересечения и касания для робота и любого из объектов. Подпространство $OBSTACLE \subseteq C$ представляет собой множество конфигураций, в которых есть хотя бы одно пересечение объектов, препятствий или робота между собой. Подпространство $FREE = C \backslash OBSTACLE$ представляет собой свободную от пересечений часть конфигурационного пространства *C* системы $Sys = Parts \cup Obs \cup Rbt$.

Геометрическая система *Sys*1, которая включает в себя робота *Rbt*, подвижный объект А и неподвижное препятствие, изображена на рис. 1.

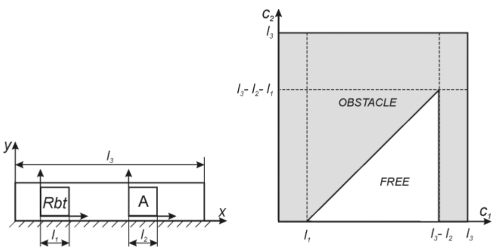

Рис. 1. Физическое и конфигурационное пространства системы *Sys*1.

3. Захват и устойчивость

Будем считать, что все трансформации элементов геометрической системы *Parts* выполняются роботом *Rbt*. Исключаются движения тел, которые вызваны любыми внешними источниками, например, свободное падение под действием силы тяжести. Пусть $c \in FREE$. Обозначим $ST(c)$ – семейство всех подмножеств тел, которые в состоянии c находятся в устойчивом положении. В семейство $ST(c)$ входят такие множества объектов, которые обладают внутренней устойчивостью. Это значит, что устойчивость данных множеств достигается только за счет относительного расположения самих объектов и неподвижных препятствий *Obs* геометрической системы *Sys*.

Обозначим $GRASP(A)$ операцию захвата роботом *Rbt* множества объектов A, а $RELEASE(A)$ операцию освобождения этого множества. Пусть $GR(c)$ – семейство всех подмножество объектов, которые может захватить робот R в конфигурации c, то есть $GR(c) = \{A | GRASP(A) \wedge Prts \backslash A \in ST(c)\}$. Множество A можно захватить, если оно обладает набором поверхностей, которые подходят для этой операции. Кроме того, исключение множества A из конфигурации c оставляет объекты из $Prts \backslash A$ в устойчивом состоянии.

Обозначим $STABLE = \{c \in FREE | \exists A \in GR(c), Prts \backslash A \in ST(c)\}$ множество точек конфигурационного подпространства *FREE*, в которых робот может осуществить захват множества объектов A (возможно пустого) и оставшиеся объекты $Prts \setminus A$ останутся в устойчивом положении.

Для любого множества объектов $A \subseteq Prts$ обозначим $P_A = \prod_{P_i \in A} P_i$ подпространство конфигурационного пространства, задающее положение объектов из A. Введем отображение $\alpha_A : C \to P_A$, которое каждой точке конфигурационного пространства ставит в соответствие конфигурацию объектов множества A.

4. Движения объектов

Движения элементов геометрической системы *Sys* можно разделить на два типа: перемещение робота без захваченного объекта (движение робота) и перемещение робота с объектом (движение объекта). Опишем формально движение робота в конфигурационном пространстве как непрерывное отображение $\tau : [0,1] \to STABLE$, для которого выполняются следующие условия:

a) $\alpha_{Prts}(\tau(0)) \in STABLE.$ В начале движения робота множество объектов *Prts* находится в стабильном состоянии;
b) $\forall x \in [0,1]\ \sigma_{Prts}(\tau(x)) = \sigma_{Prts}(\tau(0))$ В процессе движения робота все подвижные объекты остаются на месте.

В общем случае робот может перемещать сразу несколько объектов. Обозначим B такое множество объектов. Определим формально движение B в конфигурационном пространстве как непрерывное отображение $\tau: [0,1] \to STABLE$, для которого выполняются условия:

a) $\forall x \in [0,1]\; B \in Gr\big(\tau(x)\big)$. Формальное определение захвата объектов из множества B;
b) $\forall x \in [0,1]\; \pi_{Prts\backslash B}\big(\tau(x)\big) \in STABLE.$ Не захваченные объекты (множество *Prts**B*) остаются в устойчивом состоянии;
c) $\forall x \in [0,1]\; \pi_{Prts\backslash B}\big(\tau(x)\big) = \pi_{Prts\backslash B}\big(\tau(0)\big)$. Не захваченные объекты (множество *Prts**B*) неподвижны.
d) $\forall P_i \in B$ и $\forall x \in [0.1]\; E_i(\tau(x))/R(\tau(x)) = const$. Захваченные объекты занимают неподвижные позиции относительно рабочего органа (манипулятора) робота.

Преобразование множества объектов *Prts* роботом *Rbt* определим как последовательность движений в конфигурационном пространстве $t = (\tau_1, \tau_2, \dots, \tau_{2k+1})$, в которой:

a) $\forall i \in [1,2k]\;\; \tau_i(1) = \tau_{i+1}(0)$. Конечная точка одного движения служит начальной точкой следующего;
b) $\tau_1, \tau_3, \dots, \tau_{2k+1}$ – являются движениями робота *Rbt*;
c) $\tau_2, \tau_4, \dots, \tau_{2k}$ – являются движениями объектов;
d) В начале каждого движения с четным номером $\tau_{2i}, i = \overline{1,k}$ робот выполняет операцию захвата $GRASP_{2i}(A)$, где $A \in GR(\tau_{2i}(0))$;
e) В конце каждого движения с четным номером $\tau_{2i}(1), i = \overline{1,k}$ робот выполняет операцию освобождения $RELEASE_{2i}(A)$.

5. Собираемость в пространстве состояний

Полагаем, что множество объектов $Parts = \{p_1, p_2, \dots, p_n\}$ состоит из двух подмножеств $\{p_1, p_2, \dots, p_m\}$ и $\{p_{m+1}, p_{m+2}, \dots, p_n\}$. Первое подмножество образуют детали собираемого изделия, второе – технологическая оснастка, необходимая для сборки (инструменты, приспособления и др.). Обозначим $s(r)$ порядковый номер интервала $\tau_s \in t$, если существует $p_r \in \{p_1, p_2, \dots, p_m\}$ такой, что $RELEASE_s(p_r)$. Пусть p_l – базовая деталь, положение которой зафиксировано в пространстве E. Будем считать, что с этой деталью связана глобальная система координат и все отсчеты производятся от детали p_l.

Обозначим $c_i \in STABLE$ точку конфигурационного пространства, которая описывает положение всех деталей и элементов оснастки перед началом сборки. В этом положении все объекты находятся на достаточном удалении друг от друга, что формально можно описать как разделимость в трехмерном пространстве E. Пусть $c_F \in STABLE$ – точка конфигурационного пространства, которая описывает собранное изделие. Для $i = 2,3,\dots m$ обозначим $E_i(c_F)/E_l(c_F) = T_i$ – позицию, которую должна занимать деталь p_i относительно базовой детали p_l в собранном изделии, а для $i = m+1, \dots, n\; T_i$ – положение элемента оснастки на позиции хранения, рассчитанное относительно базовой детали p_l.

Сборку изделия опишем как преобразование $t = (\tau_1, \tau_{2,} \dots, \tau_{2k+1})$ множества объектов $\{p_1, \dots p_m, p_{m+1}, \dots p_n\}$ сборочным роботом *Rbt*, которое удовлетворяет условиям:

a) $\tau_1(0) = c_I$. В начальной точке конфигурационного пространства все детали и элементы оснастки находятся на исходных позициях;

b) $\tau_{2k+1}(1) = c_F$ и $\forall i = \overline{2,n}\ E_i(c_F)/E_1(c_F) = T_i$. В конечной точке c_F все детали занимают служебное положение в изделии, а элементы оснастки расположены на позициях хранения.

Последовательность сборки представляет собой вектор $\Lambda = (p_1, p_{i_2}, \ldots, p_{i_{m-1}})$ такой, что $\forall i = \overline{2, m-1}$:

a) $\exists s \in \{2,4, \ldots 2k\} | RELEASE_s(p_{l_i})$. Существует движение $\tau_s()$ детали p_{l_i}, в конечной точке $\tau_s(1)$ которого робот освобождает деталь от своего захвата;

b) $E_{l_i}(\tau_s(1))/E_1(\tau_s(1)) = T_{l_i}$. В конечной точке конфигурационного пространства деталь скоординирована относительно базовой детали согласно конструкторской документации;

c) $bound(p_{l_i}) \cap \left(\bigcup_{k \le i} bound(p_{l_i})\right) \neq \varnothing$. Вектор Λ удовлетворяет условию когерентности. Это значит, что в процессе сборки реализуются механические связи между деталями.

d) $\forall i, j \in \{2, \ldots m-1\}$, если $p_{l_i} < p_{l_j}$, то $s_{l_i} < s_{l_j}$. Упорядоченность деталей в Λ сохраняет порядок движений объектов в последовательности $(\tau_2, \tau_4, \ldots \tau_{2k})$.

Данная модель достаточно точно описывает аспекты сборочного процесса, зависящие от геометрических свойств деталей и позиционных механических связей, которые образуют структуру изделия. Геометрия изделия и механические связи между деталями – это фундаментальные конструктивные ограничения, которые нельзя нарушить. От удовлетворения этих ограничений зависит само существование проектного решения.

Модель дает возможность использовать глубоко развитые методы и алгоритмы планирования перемещений для синтеза траекторий движения деталей при сборке сложных изделий в условиях ограниченного геометрического доступа.

Список литературы

1. Божко А.Н. Методы анализа геометрической разрешимости при сборке изделий // Интернет-журнал НАУКОВЕДЕНИЕ. ЭЛ № ФС77-60397. 2016. Т. 8, №5. DOI:10.15862/82TVN516.
2. Божко А.Н., Родионов С.В. Методы искусственного интеллекта в автоматизированном проектировании процессов сборки // Наука и образование. МГТУ им. Н.Э. Баумана. Электрон. журн. ЭЛ № ФС 77-48211. 2016. № 8. DOI:10.7463/0816.0844719.
3. Ghandi S., Masehian El. Review and taxonomies of assembly and disassembly path planning problems and approaches // Computer-Aided Design. 2015. Vol. 67-68. P. 58-86. DOI:10.1016/j.cad.2015.05.001.
4. Latombe J-C. Robot motion planning. New York: Kluwer Academic Publishers, 1991. 651 p.
5. Lozano-Perez T. Wilson R. H. Assembly sequencing for arbitrary motions // Robotics and Automation. Proceedings 1993 IEEE International Conference. 1993. Vol. 2. P. 527-532. DOI:10.1109/ROBOT.1993.291904.