

УДК 531.396, 681.5.015

С. Л. К р у т и к о в

РЕКУРСИВНЫЙ МЕТОД ПОИСКА БАЗОВЫХ ИНЕРЦИОННЫХ ПАРАМЕТРОВ МАНИПУЛЯЦИОННЫХ МЕХАНИЗМОВ

Приведены основные идеи метода поиска базовых инерционных параметров, предложен рекурсивный подход к его реализации, эффективный с точки зрения быстродействия, а также рассмотрены практические аспекты применения этого подхода. Представлены результаты использования программы, разработанной согласно предложенному алгоритму, для манипулятора PUMA 560.

E-mail: serrk@yandex.ru

Ключевые слова: базовые инерционные параметры, динамическая модель манипулятора, уравнения движения в форме Лагранжа, манипулятор PUMA 560.

Знание уравнений динамики манипуляционных роботов необходимо как для применения современных моментных способов управления, так и для моделирования его движений. Описание динамики с помощью обычных¹ инерционных параметров оказывается неоднозначным: одному и тому же уравнению движения соответствует бесконечное число наборов значений таких параметров. Это приводит к появлению избыточных вычислительных операций при решении прямой и обратной задач динамики, а также не позволяет экспериментально осуществить параметрическую идентификацию уравнений движения.

Управление, а зачастую и моделирование, является задачей реального времени, т.е. необходимо выполнять достаточно непростые расчеты десятки, а то и сотни раз в секунду. Использование менее мощных вычислительных устройств, способных выполнить те же расчеты за требуемое время, позволит упростить и удешевить систему. Также существуют задачи, при решении которых не обойтись без идентификации. Вот лишь некоторые из них: создание программных имитаторов существующих манипуляторов² [1], разработка новых систем управления для старых роботов, моментное управление манипулятором с адаптацией его динамической модели.

¹Имеются в виду 10 независимых параметров для каждого звена: масса, координаты центра масс и осевые и центробежные моменты инерции. В дальнейшем будем называть их *элементарными инерционными параметрами*.

²Такие системы могут применяться, например, для тестирования системы управления роботом или для обучения операторов.

Для разрешения неоднозначности вводится множество *базовых инерционных параметров*, которое представляет собой наименьший набор параметров, определяющих уравнения динамики и соответствующих им взаимно-однозначно³, и в дальнейшем используется запись этих уравнений в терминах именно базовых параметров. Отдельной и весьма непростой задачей является поиск такого множества параметров и их выражение через обычные инерционные параметры, а также запись уравнений движения с помощью базовых инерционных параметров. Этим вопросам посвящено немало работ, например [2–5]. Автором был предложен альтернативный подход к решению задачи поиска базовых инерционных параметров [6], имеющий ряд преимуществ по сравнению с методами, описанными в указанных работах.

В [6] рассмотрены лишь теоретические аспекты предложенного метода. Настоящая статья посвящена вопросам его реализации, эффективной с точки зрения вычислений.

Поиск базовых инерционных параметров методом проекций. Напомним, что кинетическая и потенциальная энергия N -звенного манипулятора линейны относительно инерционных параметров⁴:

$$K = \sum_{i=1}^{10N} \frac{\partial K}{\partial p_i} p_i \quad \Pi = \sum_{i=1}^{10N} \frac{\partial \Pi}{\partial p_i} p_i,$$

где N — число звеньев манипулятора; p_i — некоторый инерционный параметр. В дальнейшем, если некоторая функция f (не обязательно скалярная) линейно зависит от ряда параметров p_i ($i = 1, \dots, n$), то частные производные вида $\partial f / \partial p_i$ будем называть *коэффициентами влияния* соответствующего параметра на эту функцию. Уравнение динамики манипулятора может быть записано с помощью уравнений Лагранжа 2-го рода

$$\left[\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\mathbf{q}}} \right) - \frac{\partial L}{\partial \mathbf{q}} \right]^T = \mathbf{Q},$$

где \mathbf{q} и $\dot{\mathbf{q}}$ — векторы обобщенных координат и скоростей, \mathbf{Q} — вектор обобщенных сил; $L = K - \Pi$ — функция Лагранжа. Обозначим вектор-строку, составленную из коэффициентов влияния элементарных инерционных параметров на функцию Лагранжа, как \mathbf{w}_L , а вектор-столбец, состоящий из самих этих параметров, как \mathbf{p} . Тогда уравнение движения примет следующий вид:

$$\left[\frac{d}{dt} \left(\frac{\partial \mathbf{w}_L}{\partial \dot{\mathbf{q}}} \right) - \frac{\partial \mathbf{w}_L}{\partial \mathbf{q}} \right] \mathbf{p} = \mathbf{Q}.$$

³При неизменных кинематических параметрах.

⁴Подробно материал данного параграфа изложен в [6].

Выражение в квадратных скобках представляет собой матрицу размера $N \times 10N$, элементы которой зависят от обобщенных координат, скоростей и ускорений. Обозначим ее W . Можно показать, что коэффициенты влияния на функцию Лагранжа являются векторами в бесконечномерном линейном пространстве функций q и \dot{q} . Однако эти векторы оказываются линейно зависимыми [4], а следовательно, линейно зависимыми будут и столбцы матрицы W , являющиеся по сути коэффициентами влияния на левую часть уравнения движения. Последнее и приводит к проблеме неоднозначности.

Отметим, что система векторов w_L задает некоторое конечномерное подпространство бесконечно-мерного пространства функций. Очевидно, что функция Лагранжа L принадлежит этому подпространству. Пусть имеется вектор-строка \tilde{w}_L размерностью $1 \times r$ ($r < 10N$), элементы которой являются линейно независимыми векторами упомянутого бесконечно-мерного пространства такими, что образуют базис $\mathcal{L}\{w_L\}$. Тогда базовые инерционные параметры можно определить как коэффициенты разложения вектора L по этому базису. Обозначая вектор, составленный из них, как \tilde{p} , можно записать $L = \tilde{w}_L \tilde{p}$. Вместе с тем $L = w_L p$. Пусть Y — матрица размера $r \times 10N$ координат векторов w_L в базисе \tilde{w}_L , т.е. $w_L = \tilde{w}_L Y$. Из условия инвариантности функции Лагранжа (а следовательно, и уравнения движения) следует, что справедливо следующее равенство $\tilde{p} = Y p$. Таким образом, число базовых инерционных параметров равно рангу системы векторов w_L , а их выражение через элементарные инерционные параметры определяется матрицей Y .

Отметим, что помимо свойства линейности, позволяющего получать оценку инерционных параметров как решение системы линейных алгебраических уравнений, кинетическая и потенциальная энергия манипулятора имеют особую структуру, которая может быть отражена в виде следующей теоремы.

Теорема (о базисном множестве). Система векторов w_L принадлежит конечномерному линейному пространству размерности $(N^2 + N) \cdot 5^\nu \cdot 3^{N-\nu} / 2 + 3^\nu \cdot 2^{N-\nu}$, задаваемому следующим множеством базисных векторов:

$$\beta = ((Q_{12} \cup Q_{21}) * Y_1 * \dots * Y_N) \cup (X_1 * \dots * X_N),$$

где $Q_{12} = \{\dot{q}_i^2 / 2, i=1 \dots N\}$, $Q_{21} = \{\dot{q}_i \dot{q}_j, i=2, \dots, N, j=1, \dots, i-1\}$,

$$Y_i = \begin{cases} \{1, \cos q_i, \sin q_i, \cos 2q_i, \sin 2q_i\}, \text{ вращ.;} \\ \{1, q_i, q_i^2\}, \text{ поступ.;} \end{cases}$$

$$X_i = \begin{cases} \{1, \cos q_i, \sin q_i\}, \text{ вращ.;} \\ \{1, q_i\}, \text{ поступ.,} \end{cases}$$

$i = 1 \dots N$, а ν — число вращательных звеньев.

Под операцией “*” понимается следующее: если $A = \{a_i, i = 1, \dots, n\}$ и $B = \{b_j, j = 1, \dots, m\}$, то $A * B = \{a_i b_j, i = 1, \dots, n, j = 1, \dots, m\}$.

Благодаря этой теореме базовые инерционные параметры могут быть найдены с помощью следующего простого алгоритма:

- поиск матрицы координат Z векторов системы w_L в конечно-мерном пространстве $\mathcal{L}\{\beta\}$;
- поиск базисных столбцов этой матрицы с помощью приведения ее к ступенчатому виду;
- составление матрицы координат \tilde{Z} базисной системы векторов \tilde{w}_L из базисных столбцов исходной матрицы координат Z ;
- вычисление матрицы Y по формуле $Y = \tilde{Z}^+ Z$, где операция «+» означает псевдообращение.

Последнее соотношение следует из условия инвариантности функции Лагранжа. Пусть b — вектор-строка, составленная из элементов множества β так, чтобы $w_L = bZ$ и $\tilde{w}_L = b\tilde{Z}$. Тогда справедливо равенство $Zp = \tilde{Z}\tilde{p}$, которое можно интерпретировать как неоднородную систему линейных алгебраических уравнений относительно \tilde{p} . Нетрудно заметить, что основная матрица этой системы \tilde{Z} имеет максимальный ранг, поэтому неизвестные определяются единственным образом по формуле $\tilde{p} = \tilde{Z}^+ Zp$. Сравнивая с $\tilde{p} = Yp$, получаем искомое соотношение.

Изначально предполагалась прямая реализация предложенного алгоритма в части нахождения координат векторов w_L^i ($i = 1 \dots 10N$):

- получение выражений для кинетической и потенциальной энергии механизма с помощью какой-либо библиотеки компьютерной алгебры;
- раскрытие скобок, упрощение и приведение подобных слагаемых в этих выражениях (расширение выражения);
- экспорт в строковое представление и их лексический и синтаксический анализ, например, с помощью механизма регулярных выражений.

Однако при разработке программы в соответствии с этим подходом выяснилось, что операция расширения для указанных выражений занимает довольно длительное время. Причем при использовании различных библиотек компьютерной алгебры результат оказывается схожим. Поэтому возникает необходимость в другой, более эффективной, реализации процедуры поиска проекций. Такую процедуру можно построить, вычисляя проекции рекурсивно.

Рекурсивное вычисление проекций. Из кинематики манипуляторов известно, что матрица перехода от системы координат k -го звена к системе координат $(k - 1)$ -го звена, если они построены по алгоритму

Денавита–Хартенберга⁵, имеет вид

$$A_k = \begin{bmatrix} \cos q_k & -\sin q_k \cos \alpha_k & \sin q_k \sin \alpha_k & a_k \cos q_k \\ \sin q_k & \cos q_k \cos \alpha_k & -\cos q_k \sin \alpha_k & a_k \sin q_k \\ 0 & \sin \alpha_k & \cos \alpha_k & d_k \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

где q_k, α_k, a_k, d_k называются параметрами Денавита–Хартенберга и полностью определяют кинематику механизма. Параметры q_k или d_k являются обобщенными координатами k -го звена в случае вращательного или поступательного сочленений соответственно. Тогда матрица перехода A_k для вращательного звена может быть представлена в виде⁶

$$A_k^{rot} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & s_{\alpha_k} & c_{\alpha_k} & d_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot 1 + \begin{bmatrix} 1 & 0 & 0 & a_k \\ 0 & c_{\alpha_k} & -s_{\alpha_k} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot c_{q_k} + \begin{bmatrix} 0 & -c_{\alpha_k} & s_{\alpha_k} & 0 \\ 1 & 0 & 0 & a_k \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot s_{q_k},$$

а для поступательного как

$$A_k^{tr} = \begin{bmatrix} c_{q_k} & -s_{q_k} c_{\alpha_k} & s_{q_k} s_{\alpha_k} & a_k c_{q_k} \\ s_{q_k} & c_{q_k} c_{\alpha_k} & -c_{q_k} s_{\alpha_k} & a_k s_{q_k} \\ 0 & s_{\alpha_k} & c_{\alpha_k} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot 1 + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot d_k.$$

Отметим, что в этих представлениях матрицы, на которые умножаются скалярные коэффициенты, являются постоянными, т.е. не зависят от обобщенных координат. Тогда для любого типа сочленения матрица перехода A_k может быть представлена в виде

$$A_k = \sum_{l=1}^{m_k} \tilde{A}_k^l x_k^l, \quad (1)$$

где $x_k^l \in X_k$, $m_k = |X_k|$, $X_k = \{1, \cos q_k, \sin q_k\}$ в случае вращательного звена и $X_k = \{1, d_k\}$ в случае поступательного звена, а \tilde{A}_k^l — матрицы при соответствующих x_k^l . Скалярные коэффициенты x_k^l в выражении (1) можно рассматривать как векторы в линейном пространстве не-

⁵Подробнее о кинематическом описании манипуляционных роботов с помощью однородных координат см. [7].

⁶Буквами c и s для краткости обозначены функции \cos и \sin . Нижний индекс является их аргументом.

прерывных функций, определенных на R . Очевидно, что эти векторы линейно независимы и образуют базис подпространства $\mathcal{L}\{X_k\}$.

Матрица перехода от системы координат k -го звена к абсолютной системе координат определяется соотношением $T_k = A_1 \dots A_k$. С учетом выражения (1) это соотношение запишется в виде

$$T_k = \left(\sum_{l_1=1}^{m_1} \tilde{A}_1^{l_1} x_1^{l_1} \right) \dots \left(\sum_{l_k=1}^{m_k} \tilde{A}_k^{l_k} x_k^{l_k} \right).$$

Раскрывая скобки, получаем

$$T_k = \sum_{l_1=1}^{m_1} \dots \sum_{l_k=1}^{m_k} \left(\tilde{A}_1^{l_1} \dots \tilde{A}_k^{l_k} \right) x_k^{l_k} \dots x_1^{l_1}.$$

Обозначим выражение в круглых скобках как $\tilde{T}_k^{l_1 \dots l_k}$. Тогда имеем

$$T_k = \sum_{l_1=1}^{m_1} \dots \sum_{l_k=1}^{m_k} \tilde{T}_k^{l_1 \dots l_k} x_k^{l_k} \dots x_1^{l_1}. \quad (2)$$

Из определения матрицы перехода T_k следует, что имеет место рекуррентное соотношение $T_k = T_{k-1} A_k$. Тогда с учетом выражений (1) и (2) это соотношение запишется в виде

$$T_k = \left(\sum_{l_1=1}^{m_1} \dots \sum_{l_{k-1}=1}^{m_{k-1}} \tilde{T}_{k-1}^{l_1 \dots l_{k-1}} x_{k-1}^{l_{k-1}} \dots x_1^{l_1} \right) \left(\sum_{l_k=1}^{m_k} \tilde{A}_k^{l_k} x_k^{l_k} \right).$$

Раскрыв скобки, получим

$$T_k = \sum_{l_1=1}^{m_1} \dots \sum_{l_{k-1}=1}^{m_{k-1}} \sum_{l_k=1}^{m_k} \left(\tilde{T}_{k-1}^{l_1 \dots l_{k-1}} \tilde{A}_k^{l_k} \right) x_k^{l_k} x_{k-1}^{l_{k-1}} \dots x_1^{l_1}.$$

Сравнивая полученное выражение с (2), можно заметить, что

$$\tilde{T}_k^{l_1 \dots l_{k-1} l_k} = \tilde{T}_{k-1}^{l_1 \dots l_{k-1}} \tilde{A}_k^{l_k}, \quad k = 1, \dots, N. \quad (3)$$

Для инициализации рекурсивной процедуры необходимо принять $\tilde{T}_0 = E$.

Теперь запишем выражение для коэффициента влияния l -го инерционного параметра k -го звена на потенциальную энергию механизма. Напомним, что рассматривается только потенциальная энергия сил веса [6]

$$\Pi = - [\mathbf{g}^T \ 0] \sum_{k=1}^N T_k H_k [0 \ 0 \ 0 \ 1]^T.$$

Здесь H_k — матрица инерции k -го звена манипулятора⁷, а \mathbf{g} — вектор ускорения свободного падения.

⁷Подробнее об описании инерционных свойств звеньев манипуляторов см. [7].

Учитывая соотношение (2), имеем

$$\frac{\partial \Pi}{\partial p_k^l} = - \sum_{l_1=1}^{m_1} \dots \sum_{l_k=1}^{m_k} \left([\mathbf{g}^T \ 0] \tilde{T}_k^{l_1 \dots l_k} D H_k^l [0 \ 0 \ 0 \ 1]^T \right) x_k^{l_k} \dots x_1^{l_1}, \quad (4)$$

где p_k^l — l -й элементарный инерционный параметр k -го звена манипулятора; $D H_k^l = \partial H_k / \partial p_k^l$. Очевидно, что $D H_i^l = D H_j^l \ \forall i, j \in N$, поэтому в дальнейшем будем опускать индекс звена. Обозначая выражение в круглых скобках как $D P_{10(k-1)+l}^{l_1 \dots l_k}$, а также имея ввиду, что $x_i^1 = 1$ ($i = 1, \dots, N$), получаем

$$\frac{\partial \Pi}{\partial p_k^l} = - \sum_{l_1=1}^{m_1} \dots \sum_{l_k=1}^{m_k} D P_{10(k-1)+l}^{l_1 \dots l_k} x_1^{l_1} \dots x_k^{l_k} x_{k+1}^1 \dots x_N^1.$$

Таким образом, величина $D P_{10(k-1)+l}^{l_1 \dots l_k}$ есть проекция на соответствующий базисный вектор коэффициента влияния $(10(k-1)+l)$ -го элементарного инерционного параметра на потенциальную энергию механизма.

Из динамики манипуляторов известно, что кинетическая энергия является квадратичной формой относительно обобщенных скоростей, коэффициенты которой определяются выражениями [7]

$$a_{ij} = \sum_{k=\max(i,j)}^N \text{tr} (U_{kj}^T U_{ki} H_k), \quad i, j = 1, \dots, N,$$

где величины U_{ki} являются частными производными вида $\partial T_k / \partial q_i$. Поскольку матрица квадратичной формы является симметрической, ее элементы a_{ij} и a_{ji} равны. Тогда можно вычислить только коэффициенты a_{ij} , лежащие на главной диагонали и ниже нее (т.е. с индексами $i = 1, \dots, N, j = 1, \dots, i$). С учетом сказанного кинетическая энергия механизма может быть записана в виде

$$K = \frac{1}{2} \sum_{i=1}^N a_{ii} \dot{q}_i^2 + \sum_{i=2}^N \sum_{j=1}^{i-1} a_{ij} \dot{q}_i \dot{q}_j, \quad (5)$$

причем

$$a_{ij} = \sum_{k=i}^N \text{tr} (U_{kj}^T U_{ki} H_k), \quad i = 1, \dots, N, j = 1, \dots, i. \quad (6)$$

Рассмотрим величины $U_{ki} = \partial T_k / \partial q_i$. Поскольку $T_k = A_1 \dots A_k$ и $A_i = A_i(q_i)$, то справедливо следующее:

$$U_{ki} = \begin{cases} 0, & k < i; \\ A_1 \dots \partial A_i / \partial q_i \dots A_k, & k \geq i. \end{cases}$$

Из кинематики известно, что $\partial A_i / \partial q_i = D_i A_i$ [7], где

$$D_i = \begin{cases} D_{rot}, & \sigma_i = 1, \\ D_{tr}, & \sigma_i = 0, \end{cases} \quad D_{rot} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad D_{tr} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Значение σ_i определяет тип сочленения между i -м и $(i-1)$ -м звеньями: единица соответствует вращательной паре, нуль — поступательной. Тогда с учетом (1) имеем

$$\frac{\partial A_i}{\partial q_i} = \sum_{l_i=1}^{m_i} \left(D_i \tilde{A}_i^{l_i} \right) x_i^{l_i}.$$

Подставляя это соотношение в выражение для U_{ki} и раскрывая скобки, получаем:

$$U_{ki} = \sum_{l_1=1}^{m_1} \dots \sum_{l_k=1}^{m_k} \left(\tilde{A}_1^{l_1} \dots D_i \tilde{A}_i^{l_i} \dots \tilde{A}_k^{l_k} \right) x_k^{l_k}, \dots, x_1^{l_1}.$$

Обозначим выражение в круглых скобках как $\tilde{U}_{ki}^{l_1 \dots l_k}$. Тогда последнее соотношение запишется в виде

$$U_{ki} = \sum_{l_1=1}^{m_1} \dots \sum_{l_k=1}^{m_k} \tilde{U}_{ki}^{l_1 \dots l_k} x_k^{l_k} \dots x_1^{l_1}. \quad (7)$$

С учетом того, что $T_k = T_{k-1}(q_1, \dots, q_{k-1})A_k(q_k)$, справедливы рекуррентные соотношения

$$U_{ki} = \begin{cases} U_{k-1,i} A_k, & i < k; \\ T_{k-1} D_k A_k, & i = k. \end{cases} \quad (8)$$

Действуя аналогично процедуре вывода соотношений (3), получаем

$$\tilde{U}_{ki}^{l_1 \dots l_k} = \tilde{U}_{k-1,i}^{l_1 \dots l_{k-1}} \tilde{A}_k^{l_k}, \quad i < k, \quad (9a)$$

$$\tilde{U}_{kk}^{l_1 \dots l_k} = \tilde{T}_{k-1}^{l_1 \dots l_{k-1}} D_k \tilde{A}_k^{l_k}, \quad i = k. \quad (9b)$$

Рассмотрим теперь матричное произведение $U_{kj}^T U_{ki}$. Воспользовавшись равенством (7), будем иметь

$$U_{kj}^T U_{ki} = \left(\sum_{l'_1=1}^{m_1} \dots \sum_{l'_k=1}^{m_k} \tilde{U}_{kj}^{l'_1, \dots, l'_k} x_k^{l'_k}, \dots, x_1^{l'_1} \right)^T \left(\sum_{l_1=1}^{m_1} \dots \sum_{l_k=1}^{m_k} \tilde{U}_{ki}^{l_1 \dots l_k} x_k^{l_k}, \dots, x_1^{l_1} \right).$$

Раскрыв скобки, получим

$$U_{kj}^T U_{ki} = \sum_{l'_1=1}^{m_1} \sum_{l_1=1}^{m_1} \dots \sum_{l'_k=1}^{m_k} \sum_{l_k=1}^{m_k} \left[\left(\tilde{U}_{kj}^{l'_1 \dots l'_k} \right)^T \tilde{U}_{ki}^{l_1 \dots l_k} \right] x_k^{l'_k} x_k^{l_k}, \dots, x_1^{l'_1} x_1^{l_1}. \quad (10)$$

Можно видеть, что система векторов $X_k^2 = \{x_k^i \cdot x_k^j : x_k^i, x_k^j \in X_k, i, j = 1, \dots, m_k\}$ линейно зависима. Действительно, хотя бы ввиду коммутативности рассматриваемой операции умножения множество X_k^2 будет содержать одинаковые элементы, что неизбежно ведет к линейной зависимости. Пусть система векторов Y_k образует базис пространства $\mathcal{L}\{X_k^2\}$. Тогда векторы этого пространства могут быть представлены в виде линейной комбинации базисных векторов $y_k \in Y_k$, т.е.

$$\sum_{l'=1}^{m_k} \sum_{l=1}^{m_k} (\alpha_{l'} \beta_l) x_k^{l'} x_k^l = \sum_{m=1}^{r_k} \gamma_m y_k^m, \quad (11)$$

где $r_k = |Y_k|$. Легко показать, что в качестве базисного множества пространства $\mathcal{L}\{X_k^2\}$ может использоваться $Y_k = \{1, \cos q_k, \sin q_k, \cos 2q_k, \sin 2q_k\}$ в случае вращательного звена и $Y_k = \{1, d_k, d_k^2\}$ в случае поступательного звена.

Найдем, как выражаются координаты γ_m произвольного вектора пространства $\mathcal{L}\{X_k^2\}$ в базисе Y_k через коэффициенты $\alpha_{l'}$ и β_l . Выполняя суммирование, подставляя вместо $x_k^l, x_k^{l'}$ и y_k^m соответствующие элементы X_k и Y_k и приводя подобные слагаемые, получаем следующие соотношения:

$$\begin{aligned} \gamma_1 &= \alpha_1 \beta_1 + \frac{1}{2} \alpha_2 \beta_2 + \frac{1}{2} \alpha_3 \beta_3; \\ \gamma_2 &= \alpha_1 \beta_2 + \alpha_2 \beta_1; \\ \gamma_3 &= \alpha_1 \beta_3 + \alpha_3 \beta_1; \\ \gamma_4 &= \frac{1}{2} \alpha_2 \beta_2 - \frac{1}{2} \alpha_3 \beta_3; \\ \gamma_5 &= \frac{1}{2} \alpha_2 \beta_3 + \frac{1}{2} \alpha_3 \beta_2 \end{aligned} \quad (12a)$$

— для вращательного сочленения;

$$\begin{aligned} \gamma_1 &= \alpha_1 \beta_1; \\ \gamma_2 &= \alpha_1 \beta_2 + \alpha_2 \beta_1; \\ \gamma_3 &= \alpha_2 \beta_2 \end{aligned} \quad (12б)$$

— для поступательного сочленения.

Замечание. Выражения базисных векторов \mathbf{b}_i могут быть использованы в расчетах левой части уравнения движения при решении обратной задачи динамики или энергии манипулятора при идентификации инерционных параметров. Поэтому в качестве базисного множества для вращательных звеньев лучше использовать

$$Y_k = \{1, \cos q_k, \sin q_k, \cos^2 q_k, \sin q_k \cos q_k\}.$$

Это связано с тем, что с точки зрения быстродействия эффективнее перемножить уже вычисленные значения синуса и косинуса, нежели заново вычислять эти функции от двойного аргумента. В этом случае коэффициенты преобразования (11) будут вычисляться следующим образом:

$$\begin{aligned}\gamma_1 &= \alpha_1\beta_1 + \alpha_3\beta_3; \\ \gamma_2 &= \alpha_1\beta_2 + \alpha_2\beta_1; \\ \gamma_3 &= \alpha_1\beta_3 + \alpha_3\beta_1; \\ \gamma_4 &= \alpha_2\beta_2 - \alpha_3\beta_3; \\ \gamma_5 &= \alpha_2\beta_3 + \alpha_3\beta_2.\end{aligned}$$

Можно показать, что подобные правила справедливы и в случае матричных коэффициентов:

$$\sum_{l'=1}^{m_k} \sum_{l=1}^{m_k} (A_{l'}CB_l) x_k^{l'} x_k^l = \sum_{m=1}^{r_k} G_m y_k^l. \quad (13)$$

В этом случае слагаемые вида $\alpha_{l'}\beta_l$ необходимо заменить на $A_{l'}CB_l$. Также следует помнить о некоммутативности матричного умножения.

Применяя преобразование (13) в равенстве (10) последовательно k раз, получаем

$$U_{kj}^T U_{ki} = \sum_{l_1=1}^{r_1} \cdots \sum_{l_k=1}^{r_k} \hat{U}_{kji}^{l_1 \dots l_k} y_k^{l_k} \cdots y_1^{l_1}. \quad (14)$$

Замечание. Приведение равенства (10) к форме (14) необходимо для соответствия теореме о базисном множестве, ведь исходное представление не является линейной комбинацией линейно независимых векторов.

В соответствии с соотношением (5) выражение для коэффициента влияния l -го инерционного параметра k -го звена на кинетическую энергию механизма, очевидно, имеет вид

$$\frac{\partial K}{\partial p_k^l} = \frac{1}{2} \sum_{i=1}^N \frac{\partial a_{ii}}{\partial p_k^l} \dot{q}_i^2 + \sum_{i=2}^N \sum_{j=1}^{i-1} \frac{\partial a_{ij}}{\partial p_k^l} \dot{q}_i \dot{q}_j.$$

Введем множество $V = Q_{12} \cup Q_{21}$. Нетрудно заметить, что $|V| = N(N+1)/2$. Вместе с тем $V = \{v_i\}$. Пусть множество V упорядочено так, что если $k = i(i-1)/2 + j$, то $v_k = \dot{q}_i^2/2$ при $j = i$ и $v_k = \dot{q}_i \dot{q}_j$ при $j \neq i$. Тогда

$$\frac{\partial K}{\partial p_k^l} = \sum_{i=1}^N \sum_{j=1}^i \frac{\partial a_{ij}}{\partial p_k^l} v^{i(i-1)/2+j}.$$

Проанализировав (6), можно сделать вывод о том, что $\partial a_{ij}/\partial p_k^l = 0$

при $k < i$. Учитывая выражение (14) и принимая во внимание, что $y_i^1 = 1$ ($i = 1, \dots, N$), при $k = i, \dots, N$ будем иметь

$$\frac{\partial a_{ij}}{\partial p_k^l} = \sum_{l_1=1}^{r_1} \dots \sum_{l_k=1}^{r_k} \text{tr} \left(\hat{U}_{kji}^{l_1 \dots l_k} D H^l \right) y_1^{l_1} \dots y_k^{l_k} y_{k+1}^1 \dots y_N^1.$$

Обозначая след матричного выражения в круглых скобках как $DK_{10(k-1)+l}^{ijl_1 \dots l_k}$, окончательно получаем

$$\frac{\partial K}{\partial p_k^l} = \sum_{i=1}^N \sum_{j=1}^i \sum_{l_1=1}^{r_1} \dots \sum_{l_k=1}^{r_k} DK_{10(k-1)+l}^{ijl_1 \dots l_k} y_1^{l_1} \dots y_k^{l_k} y_{k+1}^1 \dots y_N^1 v^{i(i-1)/2+j}.$$

Таким образом, величина $DK_{10(k-1)+l}^{ijl_1 \dots l_k}$ является проекцией на соответствующий базисный вектор коэффициента влияния $(10(k-1)+l)$ -го элементарного инерционного параметра на кинетическую энергию манипулятора.

Замечание. В дальнейшем для единообразия будем полагать, что индекс $ijl_1 \dots l_N$ полностью определяет любой базисный вектор из множества β . Причем при $i = 1, \dots, N, j = 1, \dots, i$ этот индекс задает вектор $v^{l_0} y_1^{l_1}, \dots, y_N^{l_N}$ ($l_0 = i(i-1)/2 + j$), соответствующий кинетической энергии, а при $i = j = 0$ – вектор $x_1^{l_1}, \dots, x_N^{l_N}$, соответствующий потенциальной энергии. Ясно, что l_k ($k = 1, \dots, N$) принимает значения в диапазоне от 1 до $|Y_k|$ в первом случае и от 1 до $|X_k|$ во втором случае.

Получим теперь рекуррентные соотношения для величин $\hat{U}_{kji}^{l_1 \dots l_k}$. Учитывая выражение (8), можно записать

$$U_{kj}^T U_{ki} = \begin{cases} A_k^T U_{k-1,j}^T U_{k-1,i} A_k, & i < k; \\ A_k^T U_{k-1,j}^T T_{k-1} D_k A_k, & i = k, \quad j < k; \\ A_k^T D_k^T T_{k-1}^T T_{k-1} D_k A_k, & i = k, \quad j = k. \end{cases} \quad (15)$$

Таким образом, рекурсивное вычисление матриц $U_{kj}^T U_{ki}$ через матрицы $U_{k-1,j}^T U_{k-1,i}$ возможно лишь в случае $i < k$. Получим эти соотношения, записав первое из равенств (15) с учетом (14) и (1):

$$U_{kj}^T U_{ki} = \left(\sum_{l'_k=1}^{m_k} \tilde{A}_k^{l'_k} x_k^{l'_k} \right)^T \times \\ \times \left(\sum_{l_1=1}^{r_1} \dots \sum_{l_{k-1}=1}^{r_{k-1}} \hat{U}_{k-1,j}^{l_1 \dots l_{k-1}} y_{k-1}^{l_{k-1}} \dots y_1^{l_1} \right) \left(\sum_{l_k=1}^{m_k} \tilde{A}_k^{l_k} x_k^{l_k} \right).$$

Раскрывая скобки и меняя порядок суммирования, получаем

$$U_{kj}^T U_{ki} = \sum_{l_1=1}^{r_1} \dots \sum_{l_{k-1}=1}^{r_{k-1}} \left(\sum_{l'_k=1}^{m_k} \sum_{l_k=1}^{m_k} \left[\left(\tilde{A}_k^{l'_k} \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^{l_k} \right] x_k^{l'_k} x_k^{l_k} \right) y_{k-1}^{l_{k-1}} \dots y_1^{l_1}.$$

Сравнивая последнее выражение с (13), можно сделать вывод, что

$$\sum_{l_k=1}^{r_k} \hat{U}_{kji}^{l_1 \dots l_k} y_k^{l_k} = \sum_{l'_k=1}^{m_k} \sum_{l_k=1}^{m_k} \left[\left(\tilde{A}_k^{l'_k} \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^{l_k} \right] x_k^{l'_k} x_k^{l_k}.$$

Тогда, в соответствии с матричным вариантом правил (12) нахождения коэффициентов преобразования (13), матрицы $\hat{U}_{kji}^{l_1 \dots l_k}$ ($k=1, \dots, N$, $i=1, \dots, k-1$, $j=1, \dots, i$) могут быть рекурсивно найдены следующим образом:

$$\begin{aligned} \hat{U}_{kji}^{l_1 \dots l_{k-1} 1} &= \left(\tilde{A}_k^1 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^1 + \left(\tilde{A}_k^3 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^3, \\ \hat{U}_{kji}^{l_1 \dots l_{k-1} 2} &= \left(\tilde{A}_k^1 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^2 + \left(\tilde{A}_k^2 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^1, \\ \hat{U}_{kji}^{l_1 \dots l_{k-1} 3} &= \left(\tilde{A}_k^1 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^3 + \left(\tilde{A}_k^3 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^1, \\ \hat{U}_{kji}^{l_1 \dots l_{k-1} 4} &= \left(\tilde{A}_k^2 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^2 - \left(\tilde{A}_k^3 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^3, \\ \hat{U}_{kji}^{l_1 \dots l_{k-1} 5} &= \left(\tilde{A}_k^2 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^3 + \left(\tilde{A}_k^3 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^2, \end{aligned} \quad (16a)$$

— в случае вращательного звена;

$$\begin{aligned} \hat{U}_{kji}^{l_1 \dots l_{k-1} 1} &= \left(\tilde{A}_k^1 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^1, \\ \hat{U}_{kji}^{l_1 \dots l_{k-1} 2} &= \left(\tilde{A}_k^1 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^2 + \left(\tilde{A}_k^2 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^1, \\ \hat{U}_{kji}^{l_1 \dots l_{k-1} 3} &= \left(\tilde{A}_k^2 \right)^T \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \tilde{A}_k^2. \end{aligned} \quad (16b)$$

— в случае поступательного звена.

Чтобы найти рекуррентные соотношения для матриц $\hat{U}_{kji}^{l_1 \dots l_k}$ ($k=1, \dots, N$, $i=k$, $j=1, \dots, k$) перепишем равенство (10) с учетом (9)

$$\begin{aligned} U_{kj}^T U_{kk} &= \sum_{l'_1=1}^{m_1} \sum_{l_1=1}^{m_k} \dots \\ &\dots \sum_{l'_k=1}^{m_k} \sum_{l_k=1}^{m_k} \left[\left(\tilde{A}_1^{l'_1} \dots D_j \tilde{A}_j^{l'_j} \dots \tilde{A}_k^{l'_k} \right)^T \tilde{A}_1^{l_1} \dots \tilde{A}_j^{l_j} \dots D_k \tilde{A}_k^{l_k} \right] x_k^{l'_k} x_k^{l_k} \dots x_1^{l'_1} x_1^{l_1}. \end{aligned}$$

Меняя порядок суммирования и пользуясь свойством дистрибутивности, представим последнее равенство в виде

$$\begin{aligned}
 U_{kj}^T U_{kk} = & \sum_{l'_k=1}^{m_k} \sum_{l_k=1}^{m_k} \left(\tilde{A}_k^{l'_k} \right)^T \left(\sum_{l'_{k-1}=1}^{m_{k-1}} \sum_{l_{k-1}=1}^{m_{k-1}} \dots \right. \\
 & \dots \left(\sum_{l'_j=1}^{m_j} \sum_{l_j=1}^{m_j} \left(D_j \tilde{A}_j^{l'_j} \right)^T \left(\sum_{l'_{j-1}=1}^{m_{j-1}} \sum_{l_{j-1}=1}^{m_{j-1}} \dots \right. \right. \\
 & \left. \left. \dots \left(\sum_{l'_1=1}^{m_1} \sum_{l_1=1}^{m_1} \left(\tilde{A}_1^{l'_1} \right)^T \tilde{A}_1^{l_1} x_1^{l'_1} x_1^{l_1} \right) \dots \right) \tilde{A}_j^{l'_j} x_j^{l'_j} x_j^{l_j} \right) \dots \left. \right) D_k \tilde{A}_k^{l'_k} x_k^{l'_k} x_k^{l_k}.
 \end{aligned} \tag{17}$$

В соответствии с преобразованием (13) двойная сумма в круглых скобках, находящаяся на самом высоком уровне вложенности, может быть представлена как

$$\sum_{l'_1=1}^{m_1} \sum_{l_1=1}^{m_1} \left(\tilde{A}_1^{l'_1} \right)^T \tilde{A}_1^{l_1} x_1^{l'_1} x_1^{l_1} = \sum_{l_1=1}^{r_1} \hat{A}_{1jk}^{l_1} y_1^{l_1},$$

причем матричные коэффициенты $\hat{A}_{1jk}^{l_1}$ определяются по соответствующим правилам (12). Подставляя в соотношение (17) вместо левой части последнего выражения его правую часть, раскрывая скобки и меняя порядок суммирования, получаем

$$\begin{aligned}
 U_{kj}^T U_{kk} = & \sum_{l_1=1}^{r_1} \left(\sum_{l'_k=1}^{m_k} \sum_{l_k=1}^{m_k} \left(\tilde{A}_k^{l'_k} \right)^T \times \right. \\
 & \times \left(\sum_{l'_{k-1}=1}^{m_{k-1}} \sum_{l_{k-1}=1}^{m_{k-1}} \dots \left(\sum_{l'_j=1}^{m_j} \sum_{l_j=1}^{m_j} \left(D_j \tilde{A}_j^{l'_j} \right)^T \left(\sum_{l'_{j-1}=1}^{m_{j-1}} \sum_{l_{j-1}=1}^{m_{j-1}} \dots \right. \right. \right. \\
 & \left. \left. \left. \dots \left(\sum_{l'_2=1}^{m_2} \sum_{l_2=1}^{m_2} \left(\tilde{A}_2^{l'_2} \right)^T \hat{A}_{1jk}^{l_1} \tilde{A}_2^{l_2} x_2^{l'_2} x_2^{l_2} \right) \dots \right) \tilde{A}_j^{l'_j} x_j^{l'_j} x_j^{l_j} \right) \dots \right) D_k \tilde{A}_k^{l'_k} x_k^{l'_k} x_k^{l_k} \left. \right) y_1^{l_1}.
 \end{aligned}$$

Повторив эту процедуру еще $(k - 1)$ раз, придем к следующему результату:

$$U_{kj}^T U_{kk} = \sum_{l_1=1}^{r_1} \dots \sum_{l_k=1}^{r_k} \hat{A}_{kjk}^{l_1 \dots l_k} y_k^{l_k} \dots y_1^{l_1}.$$

Сравнив его с равенством (14), можно заметить, что $\hat{U}_{kjk}^{l_1 \dots l_k} = \hat{A}_{kjk}^{l_1 \dots l_k}$. Матрицы $\hat{A}_{ijk}^{l_1 \dots l_i}$ ($i = 1, \dots, k, j = 1, \dots, k$) определяются с помощью следующей рекурсивной процедуры:

$$\begin{aligned} \hat{A}_{ijk}^{l_1 \dots l_{i-1} 1} &= \left(D_i^1 \tilde{A}_i^1\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{k-1}} D_i^2 \tilde{A}_i^1 + \left(D_i^1 \tilde{A}_i^3\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^3, \\ \hat{A}_{ijk}^{l_1 \dots l_{k-1} 2} &= \left(D_i^1 \tilde{A}_i^1\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^2 + \left(D_i^1 \tilde{A}_i^2\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^1, \\ \hat{A}_{ijk}^{l_1 \dots l_{k-1} 3} &= \left(D_i^1 \tilde{A}_i^1\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^3 + \left(D_i^1 \tilde{A}_i^3\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^1, \\ \hat{A}_{ijk}^{l_1 \dots l_{k-1} 4} &= \left(D_i^1 \tilde{A}_i^2\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^2 - \left(D_i^1 \tilde{A}_i^3\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^3, \\ \hat{A}_{ijk}^{l_1 \dots l_{k-1} 5} &= \left(D_i^1 \tilde{A}_i^2\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^3 + \left(D_i^1 \tilde{A}_i^3\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^2 \end{aligned} \quad (18a)$$

— в случае вращательного звена;

$$\begin{aligned} \hat{A}_{ijk}^{l_1 \dots l_{k-1} 1} &= \left(D_i^1 \tilde{A}_i^1\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^1; \\ \hat{A}_{ijk}^{l_1 \dots l_{k-1} 2} &= \left(D_i^1 \tilde{A}_i^1\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^2 + \left(D_i^1 \tilde{A}_i^2\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^1, \\ \hat{A}_{ijk}^{l_1 \dots l_{k-1} 3} &= \left(D_i^1 \tilde{A}_i^2\right)^T \hat{A}_{i-1, jk}^{l_1 \dots l_{i-1}} D_i^2 \tilde{A}_i^2 \end{aligned} \quad (18b)$$

— в случае поступательного звена, причем

$$D_i^1 = \begin{cases} E, & i \neq j, \\ D_{rot} \sigma_j + D_{tr} (1 - \sigma_j), & i = j, \end{cases}$$

$$D_i^2 = \begin{cases} E, & i \neq k, \\ D_{rot} \sigma_k + D_{tr} (1 - \sigma_k), & i = k. \end{cases}$$

Для инициализации процедуры следует положить $\hat{A}_{0jk} = E$.

Практическая реализация. Итак, соотношения (3), (16) и (18) определяют рекуррентный алгоритм вычисления проекций. Однако эта обратная рекурсивная схема вычислений (от N -го звена к первому) оказывается неэффективной, поскольку одни и те же матрицы \tilde{T} , \hat{U} и \hat{A} рассчитываются множество раз. Гораздо эффективнее применение прямой рекурсии, которая позволяет вычислять указанные матрицы только один раз. Правда в этом случае требуется дополнительно хранить в оперативной памяти $2 + \sum_{k=1}^N (k(k+1)/2 + k + 2)$ матриц размера 4×4 (на максимальной глубине рекурсии). Например, для $N = 6$ это значение равно 56. Если элементы матриц являются числами с плавающей точкой двойной точности (согласно IEEE 754), то дополнительное потребление памяти составит около 7 кБ. При этом для хранения матрицы координат Z в случае всех вращательных сочленений

для любого способа необходимо около 150 МБ. Таким образом, при использовании прямой рекурсии потребление памяти несущественно превышает аналогичный показатель в случае обратной рекурсии. Выигрыш же в быстродействии оказывается весьма существенным. Так, время работы программы, вычисляющей все элементы матрицы координат Z , построенной на базе обратной рекурсии, превышает аналогичный показатель для прямой рекурсии почти в 20 раз.

Рассмотрим теперь реализацию прямого рекурсивного алгоритма вычисления проекций. Обозначим преобразования (18) как τ_{fs}^σ , где индексы f , s и σ принимают значения нуль или единица. Если f или s равны нулю, то D_i^1 или D_i^2 соответственно равны единичной матрице. В противном случае они определяются указанными ранее соотношениями. При $\sigma = 1$ используются соотношения для вращательного звена, при $\sigma = 0$ — для поступательного. С учетом введенных обозначений очевидна справедливость следующих соотношений:

$$\hat{A}_{k,j,k+1}^{l_1 \dots l_k} = \tau_{00}^\sigma \left\{ \hat{A}_{k-1,j,k+1}^{l_1 \dots l_{k-1}} \right\}, \quad j = 1 \dots k - 1; \quad (19a)$$

$$\hat{A}_{k,k,k+1}^{l_1 \dots l_k} = \tau_{10}^\sigma \left\{ \hat{A}_{k-1,k,k+1}^{l_1 \dots l_{k-1}} \right\}; \quad (19б)$$

$$\hat{A}_{k,k+1,k+1}^{l_1 \dots l_k} = \tau_{00}^\sigma \left\{ \hat{A}_{k-1,k+1,k+1}^{l_1 \dots l_{k-1}} \right\}. \quad (19в)$$

Легко проверить, что матрицы $\hat{A}_{ijk}^{l_1 \dots l_i}$ ($j = i + 1, \dots, k$) одинаковы, поэтому далее будем их обозначать $\hat{A}_{i0k}^{l_1 \dots l_i}$. Индекс 0 выбран не случайно, ибо для этой матрицы $D_i^1 = D_i^2 = E$. Кроме того, проанализировав преобразования (18), можно заметить, что для любых j справедливо равенство $\hat{A}_{k-1,jk}^{l_1 \dots l_{k-1}} = \hat{A}_{k-1,j,k+1}^{l_1 \dots l_{k-1}}$. С учетом сказанного соотношения (19) запишутся в виде

$$\hat{A}_{k,j,k+1}^{l_1 \dots l_k} = \tau_{00}^\sigma \left\{ \hat{A}_{k-1,j,k}^{l_1 \dots l_{k-1}} \right\}, \quad j = 1, \dots, k - 1;$$

$$\hat{A}_{k,k,k+1}^{l_1 \dots l_k} = \tau_{10}^\sigma \left\{ \hat{A}_{k-1,0,k}^{l_1 \dots l_{k-1}} \right\};$$

$$\hat{A}_{k,0,k+1}^{l_1 \dots l_k} = \tau_{00}^\sigma \left\{ \hat{A}_{k-1,0,k}^{l_1 \dots l_{k-1}} \right\}$$

или, совмещая первое и третье равенства, получаем

$$\hat{A}_{k,j,k+1}^{l_1 \dots l_k} = \tau_{00}^\sigma \left\{ \hat{A}_{k-1,j,k}^{l_1 \dots l_{k-1}} \right\}, \quad j = 0, \dots, k - 1;$$

$$\hat{A}_{k,k,k+1}^{l_1 \dots l_k} = \tau_{10}^\sigma \left\{ \hat{A}_{k-1,0,k}^{l_1 \dots l_{k-1}} \right\}.$$

На основе вычисленных матриц можно определить величины $\hat{U}_{kjk}^{l_1 \dots l_k} = \hat{A}_{kjk}^{l_1 \dots l_k}$ ($k = 1, \dots, N$, $j = 1, \dots, k$) с помощью следующих преобразований:

$$\hat{A}_{kjk}^{l_1 \dots l_k} = \tau_{01}^\sigma \left\{ \hat{A}_{k-1,j,k}^{l_1 \dots l_{k-1}} \right\}, \quad j = 1, \dots, k - 1;$$

$$\hat{A}_{kkk}^{l_1 \dots l_k} = \tau_{11}^\sigma \left\{ \hat{A}_{k-1,0,k}^{l_1 \dots l_{k-1}} \right\}.$$

Таким образом k матриц $\hat{U}_{kjk}^{l_1 \dots l_k}$ ($j = 1, \dots, k$) вычисляются с помощью k матриц $\hat{A}_{k-1,jk}^{l_1 \dots l_{k-1}}$ ($j = 0, \dots, k-1$), полученных на предыдущем, $(k-1)$ -м шаге. С помощью соотношений, аналогичных (17), можно показать, что преобразования (16) являются частным случаем преобразований (18) при $i > k$. Тогда

$$\hat{U}_{kji}^{l_1 \dots l_k} = \tau_{00}^\sigma \left\{ \hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}} \right\} \quad (i = 1, \dots, k-1, \quad j = 1, \dots, i).$$

Далее приведен псевдокод функции, реализующей описанный алгоритм вычисления проекций кинетической энергии.

function ProjKinEnergy (list, prev_u, prev_a, DH, coord_matr)

//list – список, состоящий из элементов индекса $l_1 \dots l_k$

//prev_u – массив, содержащий матрицы $\hat{U}_{k-1,ji}^{l_1 \dots l_{k-1}}$,

$i = 1, \dots, k-1, \quad j = 1, \dots, i$

//prev_a – массив, содержащий матрицы $\hat{A}_{k-1,jk}^{l_1 \dots l_{k-1}}, j = 0, \dots, k-1$

n := число звеньев манипулятора;

k := номер текущего звена;

Создать массив cur_u матриц 4×4 из size(prev_u)+k элементов;

//cur_u содержит матрицы $\hat{U}_{kji}^{l_1 \dots l_k}, i = 1, \dots, k, \quad j = 1, \dots, i$

Создать массив cur_a матриц 4×4 из k+1 элемента;

//cur_a содержит матрицы $\hat{A}_{kj,k+1}^{l_1 \dots l_k}, j = 0, \dots, k$

sigma := TRUE, если $\sigma_k = 1$, и FALSE, если $\sigma_k = 0$;

m := | Y_k |;

for l := 1 **to** m **do**

 Добавить l в конец list;

 r := 0;

 //Вычисление $\hat{U}_{kji}^{l_1 \dots l_k}, i = 1..k-1, j = 1..i$

for i := 1 **to** k-1 **do**

for j := 1 **to** i **do**

 cur_u[r] := tau (l, prev_u[r], FALSE, FALSE, sigma);

 r := r + 1;

end for

end for

 //Вычисление $\hat{U}_{kjk}^{l_1 \dots l_k}, j = 1 \dots k$

for i := 1 **to** k-1 **do**

 cur_u[r] := tau (l, prev_a[i], FALSE, TRUE, sigma);

 r := r + 1;

end for

 cur_u[r] = tau (l, prev_a[0], TRUE, TRUE, sigma);


```

//Вычисление  $\hat{A}_{k,j,k+1}, j = 0..k$ 
for i := 0 to k-1 do
    cur_a[i] := tau (l, prev_a[i], FALSE, FALSE, sigma);
end for
cur_a[k] := tau (l, prev_a[0], TRUE, FALSE, sigma);
//Вычисление проекций
r := 0;
for i := 1 to k do
    for j := 1 to i do
        for p := 1 to 10 do
            row := idx_to_row_number (list, i, j);
            col := 10 * (k - 1) + p;
            coor_matr[row, col] := trace (cur_u[r] * DH[p]);
        end for
        r := r + 1;
    end for
end for
//Рекурсивный вызов функции, если текущее звено не
последнее
if k != n
    ProjKinEnergy (list, cur_u, cur_a, DH, coor_matr);
end if
Убрать из list последний элемент;
end for

```

end func

Функция `idx_to_row_number()` определяет строку матрицы координат Z , соответствующую текущему базисному вектору, задаваемому индексом $ijl_1 \dots l_k 1 \dots 1$. Мы не будем останавливаться на деталях реализации этой функции, поскольку это несущественно с точки зрения принципа работы рассматриваемого алгоритма. Псевдокод функции `tau()`, осуществляющей преобразование τ_{fs}^σ , не приводится ввиду его очевидности.

Прямая рекурсивная схема вычисления проекций потенциальной энергии не вызывает затруднений и может быть реализована следующим образом:

function ProjPotEnergy (list, prev_t, DH, coor_matr)

//list — то же, что и ранее, prev_t — матрица $\tilde{T}_{k-1}^{l_1 \dots l_{k-1}}$

n := число звеньев манипулятора;

k := номер текущего звена;

g := $[g_x \ g_y \ g_z \ 0]$;

u := $[0 \ 0 \ 0 \ 1]^T$;

m := $|X_k|$;

```

for l := 1 to m do
    Добавить l в конец list;
    a :=  $\tilde{A}_k^{l_k}$ ;
    //Вычисление  $\tilde{T}_k^{l_1 \dots l_k}$ 
    cur_t := prev_t * a;
    //Вычисление проекций
    for p := 7 to 10 do
        row := idx_to_row_number (list, 0, 0);
        col := 10 * (k - 1) + p;
        coor_matr[row, col] := g * cur_t * DH[p] * u;
    end for
    //Рекурсивный вызов функции, если текущее звено не
    последнее
    if k != n
        ProjPotEnergy (list, cur_t, DH, coor_matr);
    end if
    Убрать из list последний элемент;
end for
end func

```

Тогда процедура вычисления проекций полностью может быть выполнена с помощью функции, псевдокод которой представлен далее.

function CalcProjections (coor_matr)

```

    Создать пустой список list целых чисел;
    Создать пустой массив u матриц размера  $4 \times 4$ ;
    Создать массив a матриц размера  $4 \times 4$  из одного элемента;
    Создать массив DH матриц размера  $4 \times 4$  из 10 элементов;
    a[0] :=  $E_4$ ;
    t :=  $E_4$ ;
    for l := 1 to 10 do
        DH[l] :=  $DH^l$ ;
    end for
    ProjKinEnergy (list, u, a, DH, coor_matr);
    ProjPotEnergy (list, t, DH, coor_matr);

```

end func

Матрица координат coor_matr, заполняемая в процессе работы этой функции, в случае небольшого числа звеньев (не более четырех) хранится в оперативной памяти вычислительной машины. В противном случае она представляется в виде файла базы данных, расположенного на жестком диске вследствие своего большого размера. Так, например,

для пятизвенного механизма с вращательными сочленениями это значение равно 47118×50 . Для приведения матрицы координат к ступенчатому виду применяется модифицированный алгоритм исключения Гаусса с частичным выбором ведущего элемента⁷.

Из определения матриц \tilde{A} следует, что они содержат много нулевых элементов. Это, очевидно, справедливо и для остальных матричных коэффициентов, вычисляемых на основе \tilde{A} . Поэтому еще одним источником повышения быстродействия является специализированное представление всех этих разреженных матриц, ориентированное, главным образом, на минимизацию времени выполнения операции умножения матриц. В данной реализации использовалось упрощенное представление в виде ассоциативного массива, содержащего в качестве данных ненулевые элементы матрицы, а в качестве ключей — соответствующие им линейные индексы. Под линейным индексом здесь понимается натуральное число, вычисляемое как $l = in + j$, где i — номер строки, j — номер столбца, а n — число столбцов матрицы. Очевидно, что при известном размере матрицы номер строки и столбца определяется однозначно для каждого значения линейного индекса:

$$i = l/n, \quad j = l \% n,$$

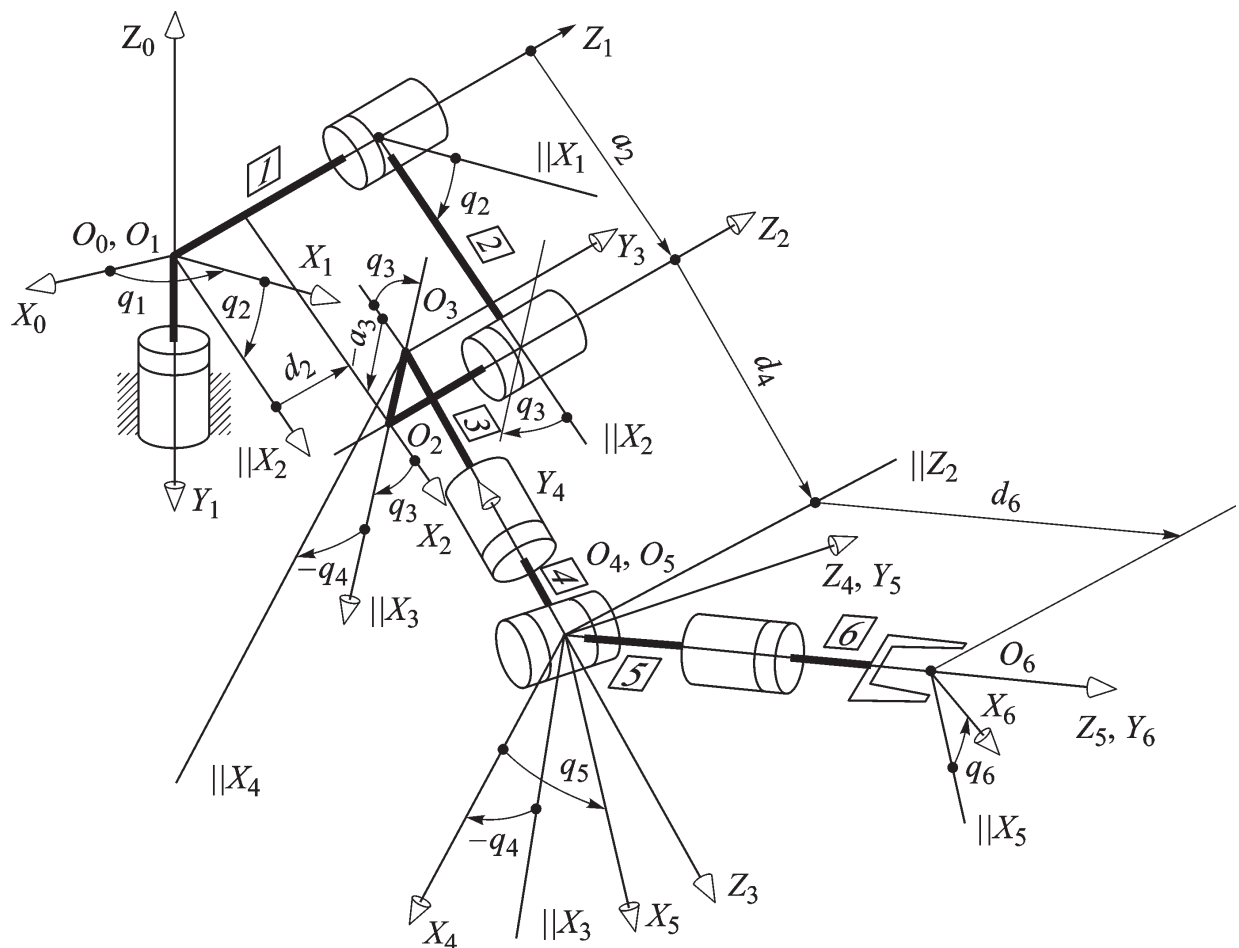
где под операцией «/» понимается целочисленное деление, а под операцией «%» — остаток от целочисленного деления. Отметим, что индексы i, j, l отсчитываются от нуля. Это позволяет исключить избыточные операции при пересчете матричных индексов в линейный и наоборот. Недостатком этой реализации является необходимость постоянного извлечения матричных индексов элемента из его линейного индекса при выполнении операции матричного умножения. Но, поскольку такая реализация применяется к разреженным матрицам небольшого размера (и соответственно с небольшим числом ненулевых элементов), заметного понижения быстродействия не происходит. Однако, если такое допущение не справедливо, необходима полноценная реализация механизма разреженных матриц с хранением обоих индексов⁸.

Пример работы приложения. Автором разработана программа на языке C++, осуществляющая поиск базовых инерционных параметров манипулятора, если заданы его кинематические параметры⁹. Создан-

⁷Этот алгоритм является стандартным и широко распространен в матричных вычислениях, поэтому в настоящей статье он не описывается. Подробнее об алгоритме см, например, в работах [8, 9].

⁸См., например, [10].

⁹В случае манипулятора с неветвящейся кинематической схемой достаточно задать типы сочленений и параметры Денавита–Хартенберга.



Кинематическая схема манипулятора PUMA 560

ное приложение основано на методе проекций и использует предложенный в данной статье рекурсивный подход к их вычислению. Эта часть программы полностью реализована согласно описанному алгоритму. Проиллюстрируем корректность работы приложения на примере хорошо известного манипулятора PUMA 560. Кинематическая схема робота представлена на рисунке, а геометрические параметры — в таблице.

Геометрические параметры манипулятора PUMA 560

№	σ_i	q_i	α_i , град	a_i , мм	d_i , мм
1	1	<i>var</i>	-90	0	0
2	1	<i>var</i>	0	431,8	150
3	1	<i>var</i>	90	-20,3	0
4	1	<i>var</i>	-90	0	433,1
5	1	<i>var</i>	90	0	0
6	1	<i>var</i>	0	0	70

Вводя параметры из этой таблицы в программу и запуская ее на исполнение, получаем в результате следующие базовые инерционные

параметры¹⁰:

$$\begin{aligned}
 \tilde{p}_1 &= I_{xx}^1 + I_{zz}^1 + I_{zz}^2 + I_{yy}^3 + 300(S_z^2 + S_y^3) + \\
 &\quad + 22500(m_2 + m_3 + m_4 + m_5 + m_6) \\
 \tilde{p}_2 &= I_{zz}^3 + I_{yy}^4 - 866,2 S_y^4 + 187576(m_4 + m_5 + m_6) \\
 \tilde{p}_3 &= I_{xx}^2 - 186451(m_2 + m_3 + m_4 + m_5 + m_6) \\
 \tilde{p}_4 &= S_x^2 + 431,8(m_2 + m_3 + m_4 + m_5 + m_6) \\
 \tilde{p}_5 &= I_{xx}^3 - 412,09(m_3 + m_4 + m_5 + m_6) \\
 \tilde{p}_6 &= S_z^3 - S_y^4 + 433,1(m_4 + m_5 + m_6) \\
 \tilde{p}_7 &= S_x^3 - 20,3(m_3 + m_4 + m_5 + m_6) \\
 \tilde{p}_8 &= I_{zz}^5 + I_{zz}^6 + 140 S_z^6 + 4900 m_6 \\
 \tilde{p}_9 &= I_{xz}^2 + 431,8(S_z^2 + S_y^3) \\
 \tilde{p}_{10} &= S_z^5 + S_z^6 + 70 m_6 \\
 \tilde{p}_{11} &= I_{xy}^3 - 20,3 S_y^3 \\
 \tilde{p}_{12} &= I_{zz}^4 + I_{yy}^5 \\
 \tilde{p}_{13} &= S_z^4 + S_y^5 \\
 \\
 \tilde{p}_{14} &= I_{yy}^2 & \tilde{p}_{15} &= I_{xy}^2 & \tilde{p}_{16} &= I_{yz}^2 & \tilde{p}_{17} &= S_y^2 & \tilde{p}_{18} &= I_{xz}^3 & \tilde{p}_{19} &= I_{yz}^3 \\
 \tilde{p}_{20} &= I_{xx}^4 & \tilde{p}_{21} &= I_{xy}^4 & \tilde{p}_{22} &= I_{xz}^4 & \tilde{p}_{23} &= I_{yz}^4 & \tilde{p}_{24} &= S_x^4 & \tilde{p}_{25} &= I_{xx}^5 \\
 \tilde{p}_{26} &= I_{xy}^5 & \tilde{p}_{27} &= I_{xz}^5 & \tilde{p}_{28} &= I_{yz}^5 & \tilde{p}_{29} &= S_x^5 & \tilde{p}_{30} &= I_{xx}^6 & \tilde{p}_{31} &= I_{yy}^6 \\
 \tilde{p}_{32} &= I_{xy}^6 & \tilde{p}_{33} &= I_{xz}^6 & \tilde{p}_{34} &= I_{yz}^6 & \tilde{p}_{35} &= S_x^6 & \tilde{p}_{36} &= S_y^6
 \end{aligned}$$

Число базовых инерционных параметров и их выражения через элементарные инерционные параметры соответствуют результатам, полученным в работах [2, 3 5].

Заключение. Итак, в настоящей работе были получены соотношения, позволяющие вычислять проекции коэффициентов влияния с помощью рекурсивной процедуры. На основе этих соотношений предложен алгоритм и разработана программа, осуществляющая поиск базовых инерционных параметров в соответствии с оригинальным методом [6]. Необходимо отметить высокую сложность реализации предлагаемого алгоритма. Однако рекурсивный подход позволяет добиться высокого быстродействия при использовании метода проекций. Так полное время работы созданного приложения для робота PUMA 560 составило около 200 мс на персональном компьютере средней мощности. Это значение сравнимо с аналогичным показателем численных способов поиска базовых инерционных параметров,

¹⁰Коэффициенты при статических моментах и массах имеют размерности мм и мм² соответственно.

описанных, например, в [2]. Отметим, что эти методы принципиально являются приближенными и не дают гарантий получения точного решения в отличие от предложенного автором метода проекций. В заключение отметим, что упомянутые рекурсивные соотношения могут быть распространены на случай кинематических схем с ветвлениями и произвольных кинематических пар. В дальнейшем при обобщении предложенного метода для более широкого класса механизмов предполагается использование именно такого подхода.

СПИСОК ЛИТЕРАТУРЫ

1. Z e n k e v i c h S., N a z a r o v a A., N a g y S. Computer-based arm // Proc. of 23rd International Symposium on Industrial Robots (1992), p. 116–121.
2. G a u t i e r M. Numerical calculation of the base inertial parameters of robots // IEEE Transactions on Robotics and Automation. – 1990. Vol. 6. No. 3. – P. 368–373.
3. G a u t i e r M., K h a l i l W. Direct calculation of minimum set of inertial parameters of serial robots // IEEE Transactions on Robotics and Automation. – 1990. – Vol. 6, No. 3. – P. 368–373.
4. М а y e d a Н., Y o s h i d a К., O s u k a К. Base parameters of manipulator dynamic models // IEEE Trans. on Robotics and Automation. – 1990. – Vol. 6. No. 3. – P. 312–320.
5. S h e u S. -Y., W a l k e r M. W. Basis sets for manipulator inertial parameters // Proc. of IEEE Conf. on Robotics and Automation (1989), p. 1517–1522.
6. К р у т и к о в С. Л. Базовые инерционные параметры манипуляционных роботов. Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение. – 2010. № 1. – С. 10–20.
7. З е н к е в и ч С. Л., Ю щ е н к о А. С. Основы управления манипуляционными роботами. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2004.
8. Г о л у б Д ж., Ч. В а н Л о у н. Матричные вычисления. – М.: Мир, 1999.
9. В e e z e r R. A. A first course in linear algebra [Электронный документ]. (2004).
10. В a r r e t R., В e r r y M. et al. Templates for the solution of linear systems: Building blocks for iterative methods. PA Philadelphia, SIAM (1994).

Статья поступила в редакцию 26.01.2011

Сергей Леонидович Крутиков родился в 1985 г., окончил МГТУ им. Н.Э. Баумана в 2008 г., получил степень магистра техники и технологии по направлению "Автоматизация и управление". Ассистент кафедры "Робототехнические системы" МГТУ им. Н.Э. Баумана. Автор двух научных работ в области робототехники.

S.L. Krutikov (b. 1985) graduated from the Bauman Moscow State Technical University in 2008, Assistant of "Robotic Systems" department of the Bauman Moscow State Technical University. Author of 2 publications in the field of robotics.