

<https://doi.org/10.26160/2572-4347-2025-23-33-45>

МОДЕЛЬ ДАННЫХ ДЛЯ СИМВОЛЬНОЙ ОПТИМИЗАЦИИ АЛГОРИТМОВ МОДЕЛИРОВАНИЯ ДИНАМИКИ СИСТЕМ ТВЕРДЫХ ТЕЛ

Яскевич А.В.

*ПАО «Ракетно-космическая корпорация «Энергия» им. С.П. Королева»,
Королев, Россия*

Ключевые слова: механическая система многих тел, уравнения динамики, рекуррентные алгоритмы моделирования, вычислительная эффективность, система символьных преобразований, компьютерное представление математических выражений.

Аннотация. Алгоритмы моделирования динамики механических систем многих тел разрабатываются и используются в векторно-матричной форме, избыточной для конкретной модели. Современные рекуррентные алгоритмы являются более эффективными, чем алгоритмы, основанные на уравнениях аналитической механики. Однако для моделирования сложных механических систем в реальном времени необходима их дополнительная символьная оптимизация, исключающая избыточные вычисления. Рассматривается модель данных для компьютерного представления скалярных и матричных математических выражений в специализированной системе символьных преобразований.

A DATA MODEL FOR SYMBOLIC OPTIMIZATION OF MULTI BODY DYNAMIC SIMULATION ALGORITHMS

Yaskevich A.V.

Rocket Space Corporation “Energia” n.a. S.P. Korolev, Korolev, Russia

Keywords: multi body mechanical system, dynamic equations, recurrent simulation algorithms, computation efficiency, symbolic manipulation system, computer representation of mathematical expression.

Abstract. Dynamic simulation algorithms for multi body mechanical system are developed and used in vector-matrix form, redundant for specific models. Modern recurrent algorithms have more efficiency than algorithms based on equations of analytical mechanics. However, theirs additional symbolic optimization exclusive redundant computing needed for real time simulation of complex mechanical systems. Data model for computer representation of scalar and matrix expression in specialized symbolic manipulation system considered in this paper.

Введение. Расчет движения механических систем (МС) твердых тел осуществляется численным интегрированием уравнений динамики (УД), составленных относительно ускорений. Основными математическими объектами в этих уравнениях являются векторы положений, скоростей и ускорений, тензоры инерции тел, выраженные в декартовых системах координат, матрицы преобразования между этими системами. Структура связей тел, типы шарниров могут быть учтены неявно, описаны вспомогательными векторами или матрицами. Векторно-матричная форма записи алгоритма расчета коэффициентов УД определяет метод расчета ускорений для некоторого класса МС. При численной реализации алгоритма сборка уравнений для конкретной модели осуществляется на каждом шаге

интегрирования подстановкой в векторы и матрицы конкретных численных значений геометрических и инерционных параметров тел, учетом типов шарниров. Платой за относительную универсальность матричных алгоритмов является их вычислительная избыточность. Эффективность численного моделирования конкретной МС может быть увеличена посредством перехода от обобщенной матричной записи алгоритма к эквивалентной не избыточной совокупности скалярных математических выражений (СМВ).

Компьютерные технологии выполнения в символьном виде математических операций активно развивались со второй половины прошлого века. Обзоры многочисленных работ в данной области исследований приведены в [1, 2]. В прежних [3, 4] и новых [5] универсальных системах аналитических вычислений, ориентированных на выполнение разнообразных математических операций, формулы в компьютерной памяти представляются в виде списков – упорядоченных многоуровневых наборов данных [6]. Универсальные системы предоставляют ограниченные возможности программирования сложных матричных алгоритмов расчета движения МС.

Специализированные системы символьных преобразований (ССП) для формирования уравнений динамики систем тел [7-15] интенсивно разрабатывались вплоть до конца 20 века, в основном из-за малой производительности компьютеров того времени. Они реализовывали в символьном виде простейшие скалярные математические операции – сложение, вычитание, умножение, и на этой основе векторно-матричные алгоритмы. Последние были получены из уравнений аналитической механики. При таком подходе все векторные величины определяются в общей, базовой системе координат (СК) и движение каждого отдельного тела выражается в явном виде через относительные перемещения и скорости во всех предшествующих шарнирах. Вследствие этого СМВ, определяющие элементы векторов и матриц, являются алгебраическими полиномами относительно элементов вектора шарнирных переменных, т.е. обобщенных координат и их производных по времени. Такой постоянный состав переменных позволяет упростить кодирование выражений, но не обеспечивает исключения значительной части повторяющихся вычислений.

В конце 70-х годов были разработаны новые, рекуррентные алгоритмы, позволяющие более чем на порядок сократить объем вычислений. Они определяют движение каждого тела только через движение непосредственно ему предшествующего в локальных СК этих тел. Алгоритм составного тела, обеспечивающий расчет коэффициентов УД в замкнутой форме, был впервые предложен в [16, 17], а алгоритм присоединенного тела, позволяющий рассчитать ускорения без использования матрицы инерции, – в работах [18-20]. С учетом возможностей современных компьютеров они обеспечивают приемлемое время моделирования. Однако для расчета движения в реальном времени сложных МС необходима их дополнительная символьная оптимизация.

В этом случае основной особенностью символьной реализации является необходимость представления СМВ с произвольным составом переменных, которые обозначают результаты уже выполненных вычислений. В данной работе рассматривается новый табличный способ кодирования СМВ с произвольным составом переменных, а также способ выполнения в символьном виде скалярных и матричных операций.

1. Основные вычислительные особенности рекуррентных алгоритмов расчета ускорений. Они кратко поясняются на примере расчета вектора правой части в УД в замкнутой форме для простой кинематической цепи на основе алгоритма составного тела. Векторы и матрицы, определяющие взаимное положение локальных систем координат двух смежных тел B_{j-1} и B_j , соединенных шарниром J_j показаны на рисунке 1.

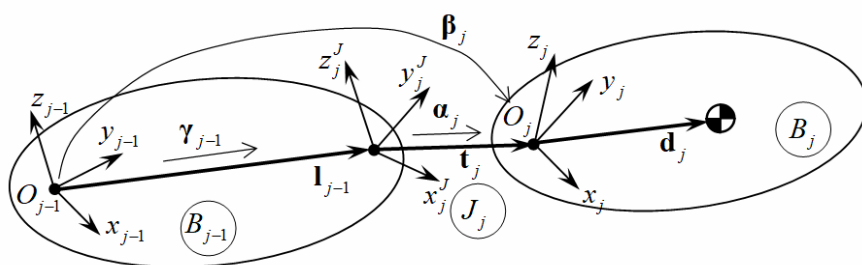


Рис. 1. Относительное положение и системы координат двух смежных тел кинематической цепи

С шарниром связаны выходная СК $O_{j-1}^J x_{j-1}^J y_{j-1}^J z_{j-1}^J$ тела B_{j-1} и входная СК $O_j x_j y_j z_j$ тела B_j . Положение $O_j^J x_{j-1}^J y_{j-1}^J z_{j-1}^J$ относительно $O_{j-1} x_{j-1} y_{j-1} z_{j-1}$ задается постоянным вектором $\mathbf{l}_{j-1}$ и постоянной матрицей преобразования γ_j . Тело B_j имеет массу m_j и тензор инерции $\mathbf{I}_j$. Положение центра масс относительно $O_j x_j y_j z_j$ задается постоянным вектором $\mathbf{d}_j$. Относительные поступательное и угловое перемещения в шарнире определяются переменным вектором $\mathbf{t}_j$ и переменной матрицей преобразования α_j из СК $O_{j-1}^J x_{j-1}^J y_{j-1}^J z_{j-1}^J$ в $O_j x_j y_j z_j$. Их элементы зависят от шарнирной переменной q_j , элемента вектора $\mathbf{q}$ обобщенных координат МС (в алгоритме составного тела рассматриваются шарниры только с одной степенью свободы). Относительные скорости $\mathbf{v}_j^{rel}$ и $\boldsymbol{\omega}_j^{rel}$ в шарнире определяются через постоянные (3×1) –матрицы $\mathbf{T}_j^{rel}$ и $\mathbf{R}_j^{rel}$ парциальных скоростей, в которых только один из элементов равен 1, остальные нулевые. В поступательном шарнире $(r_j^J = 0)$ $\mathbf{R}_j^{rel} = \mathbf{0}$, во вращательном $(r_j^J = 1)$ – $\mathbf{T}_j^{rel} = \mathbf{0}$. Преобразование из $O_{j-1} x_{j-1} y_{j-1} z_{j-1}$ в $O_j x_j y_j z_j$ определяется матрицей β_j . Внешние активные силы $\mathbf{f}_j^E$ и моменты $\mathbf{m}_j^E$ действуют на тело B_j , а

внутренние активные либо сила f_j^J , либо момент m_j^J – в шарнире J_j относительно оси движения на тела B_j и B_{j-1} (с обратным знаком).

Прямой и обратный циклы рекуррентного алгоритма расчета вектора $\mathbf{b}$ в правой части уравнений динамики в замкнутой форме $\mathbf{A}\ddot{\mathbf{q}}=\mathbf{b}$ для кинематической цепи из N тел записываются в виде следующего псевдокода

```

for  $j=1$  to  $N$  do
     $\mathbf{r}_j^{(j-1)} = \mathbf{l}_{j-1} + \boldsymbol{\gamma}_{j-1}^T \mathbf{t}_j^{rel}$ ,  $\boldsymbol{\beta}_j = \boldsymbol{\alpha}_j \boldsymbol{\gamma}_{j-1}$ 
     $\mathbf{v}_j^{rel} = \mathbf{T}_j^{rel} \dot{\mathbf{q}}_j$ ,  $\boldsymbol{\omega}_j^{rel} = \mathbf{R}_j^{rel} \dot{\mathbf{q}}_j$ ,
     $\boldsymbol{\omega}_{j-1}^{(j)} = \boldsymbol{\beta}_j \boldsymbol{\omega}_{j-1}$ ,  $\boldsymbol{\omega}_j = \boldsymbol{\omega}_{j-1}^{(j)} + \boldsymbol{\omega}_j^{rel}$ ,
     $\boldsymbol{\varepsilon}_j = \boldsymbol{\beta}_j \boldsymbol{\varepsilon}_{j-1} + \tilde{\boldsymbol{\omega}}_{j-1}^{(j)} \boldsymbol{\omega}_j^{rel}$ ,  $\mathbf{E}_j = \tilde{\boldsymbol{\varepsilon}}_j + \tilde{\boldsymbol{\omega}}_j \tilde{\boldsymbol{\omega}}_j$ ,
     $\mathbf{w}_j = \boldsymbol{\beta}_j (\mathbf{w}_{j-1} + \mathbf{E}_{j-1} \mathbf{r}_j^{(j-1)}) + \tilde{\boldsymbol{\omega}}_{j-1}^{(j)} (\mathbf{v}_j^{rel} + \mathbf{v}_j^{rel})$ ,
     $\mathbf{f}_j^{I,n} = -m_j \mathbf{w}_j - \mathbf{E}_j \mathbf{c}_j^c + \mathbf{f}_j^E$ ,
     $\mathbf{m}_j^{I,n} = -\tilde{\mathbf{c}}_j^c \mathbf{w}_j - \mathbf{I}_j \boldsymbol{\varepsilon}_j - \tilde{\boldsymbol{\omega}}_j (\mathbf{I}_j \boldsymbol{\omega}_j) + \mathbf{m}_j^E$ 
end
for  $j = N$  to  $1$  do
     $\mathbf{b}(j) = \begin{cases} \mathbf{T}_j^{rel T} \mathbf{f}_j^{I,n} + f_j^J, & r_j^J = 0, \\ \mathbf{R}_j^{rel T} \mathbf{m}_j^{I,n} + m_j^J, & r_j^J = 1, \end{cases}$ 
    if  $j > 1$  then do
         $\mathbf{f}_j^{I,n(j-1)} = \boldsymbol{\beta}_j^T \mathbf{f}_j^{I,n}$ ,
         $\mathbf{f}_{j-1}^{I,n} = \mathbf{f}_{j-1}^{I,n} + \mathbf{f}_j^{I,n(j-1)}$ ,
         $\mathbf{m}_{j-1}^{I,n} = \mathbf{m}_{j-1}^{I,n} + \boldsymbol{\beta}_j^T \mathbf{m}_j^{I,n} + \tilde{\mathbf{r}}_j^{(j-1)} \mathbf{f}_j^{I,n(j-1)}$ 
    end
end
    
```

где векторное произведение (3×1) –векторов $\mathbf{a}$ и $\mathbf{b}$ записывается с использованием кососимметрической матрицы $\tilde{\mathbf{a}}$, т.е. $\mathbf{a} \times \mathbf{b} = \tilde{\mathbf{a}} \mathbf{b}$.

Это позволяет выделить следующие основные свойства рекуррентных алгоритмов:

- векторы и матрицы являются математическими объектами, индексируемыми по номеру тела в кинематической цепи;
- отсутствуют повторяющиеся вычисления, в последующих выражениях они представлены промежуточными переменными;
- состав промежуточных переменных, т.е. способ обозначения результата вычислений, может быть произвольным;
- применение (3×1) –векторов и (3×3) –матриц уменьшает объем избыточных вычислений по сравнению с блочной векторно-матричной формой записи, в которой осуществляется вывод алгоритмов;

– операции с двумя одинаковыми векторными или матричными аргументами, в частности, $\mathbf{E}_j = \tilde{\mathbf{e}}_j + \tilde{\mathbf{\omega}}_j \tilde{\mathbf{\omega}}_j$ и $-\mathbf{I}_j \mathbf{e}_j - \tilde{\mathbf{\omega}}_j (\mathbf{I}_j \mathbf{\omega}_j)$ для повышения эффективности вычислений могут быть реализованы по специальным оптимизирующим формулам (ОФ), использующим только скалярные операции.

Компьютерная реализация индексируемых векторно-матричных выражений в несколько раз увеличивает время вычислений. В наиболее эффективной форме записи алгоритмов индексы элементов векторов и матриц вносятся в их идентификаторы, и у всех переменных остается только индекс j тела. Без учета затрат на выборку индексируемых величин приведенный выше алгоритм для кинематической цепи из 7 тел (космический манипулятор типа SSRMS [21]) требует выполнения 1205 умножений и 1072 сложений.

2. Табличное кодирование матричных математических выражений.

В специализированных ССП кодирование символьных переменных упрощает реализацию математических операций, уменьшает затраты памяти. При произвольном составе переменных информация о них хранится в таблице IDT (Identifier Description Table). Каждая ее строка содержит:

- идентификатор матрицы, вектора, массива и число символов в нем;
- тип матрицы, вектора, который обозначает идентификатор;
- размерность матрицы, как одномерного массива.

Тип матрицы определяет способ ее генерации перед выполнением символьных преобразований. Используется несколько типов матриц (массивов): внешняя исходная неиндексируемая, например, вектор $\mathbf{q}$ обобщенных координат; внешняя неиндексируемая, целевая, результат вычислений, например вектор $\mathbf{b}$; внешняя исходная индексируемая, например, вектор $\mathbf{f}_j^E$; внутренняя индексируемая постоянная, например, тензор инерции $\mathbf{I}_j$; внутренняя неиндексируемая постоянная, например, массив масс $m_j, j = \overline{1, N}$; внутренняя индексируемая переменная с элементами, зависящими от внешнего исходного вектора, например, матрица преобразования $\mathbf{a}_j$; массив промежуточных переменных; внутренняя индексируемая переменная с элементами этого массива, например, вектор угловой скорости $\mathbf{\omega}_j$. Кодами идентификаторов векторов, матриц и массива промежуточных переменных являются номера строк их описания, которые заносятся в IDT последовательно.

Векторно-матричные операции преобразуются в эквивалентную последовательность СМВ. Каждая из них определена для двух операндов. СМВ, определяющие элементы матрицы-результата, являются суммами произведений-слагаемых с двумя сомножителями. Описания этих СМВ накапливаются в двух таблицах – таблице произведений PDT (Product Description Table) и в таблице сумм SDT (Sum Description Table). Для упрощения верификации получаемого в символьном виде исходного кода,

сравнением результатов его выполнения с аналогичным численным алгоритмом, все векторные и матричные константы задаются в численном виде. Поэтому один из двух сомножителей в произведении-слагаемом может быть вещественным числом. В алгоритме присоединенного тела используется операция деления. Поэтому каждая строка таблицы PDT содержит:

- вещественный численный сомножитель;
- код идентификатора и индекс (старый и новый) первого символьного сомножителя;
- код идентификатора и индекс (старый и новый) второго символьного сомножителя;

– код идентификатора и индекс (старый и новый) символьного делителя.

Каждая строка таблицы SDT содержит

- код идентификатора и индекс («старый» и новый) суммы;
- признак использования;
- номера строк первого и последнего произведения-слагаемого в PDT.

«Старый» индекс присваивается в процессе символьной реализации векторно-матричного алгоритма. Применение признака использования и нового индекса при исключении избыточных вычислений и поясняется позднее. После выполнения всех векторно-матричных операций на основе содержимого таблиц SDT и PDT формируется в символьном виде исходный код вычислительной процедуры.

Описания всех объектов алгоритма хранятся в таблице параметров матриц MPT (Matrix Parameter Table) и в таблице элементов матриц MET (Matrix Element Table). Каждая строка первой из них содержит:

- код идентификатора и тип матрицы в соответствии с описанием в IDT;
- индекс матрицы в алгоритме;
- число строк и столбцов матрицы;
- номер первой и последней строки описания элементов в таблице MET.

Все матрицы хранятся по столбцам, для доступа к элементу i –й строки и j –го столбца используется линейный индекс $l = (j - 1) \cdot 3 + i$. Векторы имеют единственный столбец.

Описания элементов матрицы не содержат математических выражений, только численный или символьный результат операции. Каждая строка таблицы MET содержит:

- значение линейного индекса элемента в матрице;
- вещественное численное значение элемента;
- код идентификатора символьной переменной элемента и ее индекс.

Если код идентификатора равен нулю, то элемент является числом со знаком. В противном случае знак символьного элемента содержится в вещественном числе, равном ± 1 . Идентификатор элемента совпадает с идентификатором символьного вектора только если последний является внешним исходным. Во всех остальных случаях индекс элемента равен номеру строки в таблице SDT, а код идентификатора соответствует массиву

промежуточных переменных, если вектор или матрица не внешние целевые, которые заносятся в строки SDT со своим кодом идентификатора.

3. Состав системы символьных преобразований. Каждая строка приведенных выше таблиц в терминах программирования представляет собой структуру. Таблицы реализованы как файлы прямого доступа, отображаемые на память, что позволяет работать с ними как с массивами структур. Программные модули специализированной ССП написаны на языке высокого уровня и содержат процедуры, реализующие многократно используемые операции или действия. Управление последовательностью выполнения процедур при реализации конкретного векторно-матричного алгоритма программируется на том же языке высокого уровня. Отдельные модули СП выполняют следующие функции:

- создание таблиц IDT, MPT, MET, SDT, PDT для заданной модели MC;
- создание описаний векторов и матриц всех типов в IDT, MPT, MET;
- выполнение скалярных операций (скалярный символьный процессор);
- выполнение матричных операций;
- реализация оптимизирующих формул;
- реализация заданного векторно-матричного алгоритма;
- генерация в символьном виде на основе таблиц IDT, SDT, PDT исходного кода программы расчета, эквивалентной векторно-матричному численному алгоритму.

Предварительное создание описаний векторов и матриц операндов и результата в таблицах обеспечивает контроль правильности доступа к их элементам в процессе выполнения операций. Для ускорения доступа коды используемых в алгоритме идентификаторов и номера индексируемых матриц дублируются в соответствующих скалярных переменных и в массивах. Формирование исходных и постоянных векторов и матриц основано на содержимом файла описания численных параметров MC, общем для ССП и для программы, реализующей численный алгоритм.

Скалярный символьный процессор выполняет операции умножения, сложения и вычитания двух скалярных операндов, а также скалярного произведения двух (3×1) –векторов. Если результат не является числом или ранее полученной символьной переменной, т.е. является СМВ, содержащим вычисления, то он обозначается промежуточной переменной и записывается в таблицы SDT и PDT. В модулях реализации матричных вычислений или оптимизирующих формул результат операции как число или символьная переменная могут быть занесены в описание элемента соответствующей матрицы в MET.

Модуль выполнения типовых операций с (3×1) –векторами (3×3) –матрицами использует скалярный процессор. Фиксированная размерность объектов упрощает программный код и повышает его надежность. Реализованы умножение скалярной величины на вектор, матрицы на вектор, транспонированной матрицы на вектор, матрицы на матрицу,

скалярное произведение векторов, а также сложение и вычитание векторов и матриц. В выражениях с более чем двумя матричными операндами для фиксации промежуточных результатов могут использоваться временные векторы и матрицы. Модуль содержит функции очистки их описаний перед началом очередного, многократного применения.

Модуль реализации уникальных оптимизирующих формул для вычисления выражений с двумя одинаковыми векторными или матричными операндами использует только скалярные операции символьного процессора и промежуточные символьные переменные в виде отдельных структур данных, аналогичных строкам таблицы МЕТ. Результат оптимизирующей формулы представляет собой матрицу, описание которой предварительно создается в таблицах МРТ и МЕТ.

В алгоритмах расчета ускорений матрицы, зависящие только от обобщенных координат, могут вычисляться реже, чем векторы, зависящие от обобщенных скоростей и активных сил. Например, в алгоритме составного тела матрица $\mathbf{A}$ обобщенной инерции в уравнениях динамики $\mathbf{A}\ddot{\mathbf{q}}=\mathbf{b}$ в замкнутой форме, может вычисляться реже и отдельно от вектора $\mathbf{b}$ обобщенных сил в правой части. Подобное может быть реализовано и в алгоритме присоединенного тела. В этом случае для каждой из двух частей уравнений выделяется своя часть общего алгоритма. В отдельном модуле ССП она инициирует выполнение всех необходимых функций и программируется на том же языке высокого уровня, на котором написаны исходные коды других модулей. Это значительно упрощает программную реализацию. Подобные отдельные прикладные модули ССП могут также обеспечивать расчет кинематики МС по соответствующим векторно-матричным алгоритмам. Их число определяется постановкой конкретной задачи.

Модуль генерации в символьном виде исходного кода программы расчета формирует последовательность СМВ в соответствии с содержимым таблиц IDT, SDT, PDT. Он не зависит от вида оптимизируемого векторно-матричного алгоритма.

4 Выполнение операций в скалярном символьном процессоре.

Скалярные операции являются основой для реализации матричных алгоритмов. Для их выполнения в скалярном символьном процессоре имеются шесть структур данных для хранения операндов, символьных переменных $\{n_{O,i}, C_{O,i}, I_{O,i}\}, i=\overline{1,6}$; три структуры для формирования произведений-слагаемых $\{n_{P,i}, N_{SM,i}, C1_{P,i}, I1_{P,i}, C2_{P,i}, I2_{P,i}\}, i=\overline{1,3}$ и одна структура для описания результата-суммы $\{n_S, C_S, I_S\}$. Здесь и далее символ n обозначает вещественный численный множитель, символы $C, C1, C2$ и $I, I1, I2$ соответственно коды идентификаторов и индексы символьных переменных, а нижние индексы указывают на принадлежность к соответствующей структуре. Переменная N_{SM} обозначает число символьных сомножителей и упрощает алгоритм выполнения операции, то есть повышает

его надежность. Занесение операндов в символьный процессор и символьной переменной, обозначающей скалярный результат, в описания элементов матриц или промежуточных переменных оптимизирующих формул осуществляют процедуры, реализующие соответствующие операции.

Умножение двух скалярных символьных или численных операндов с использованием описанных выше структур данных выполняется в следующей последовательности.

1. Умножаются численные множители операндов $n_{p,1} = n_{o1} \cdot n_{o,2}$.
2. При $n_{p,1} = 0$ результат равен нулю, выполняется выход из символьного процессора:

$$n_{p,1} = 0 \rightarrow \{n_s = 0, C_s = 0, I_s = 0\}, \text{return};$$

3. Формируются символьные сомножители

$$C_{o,1} \neq 0 \rightarrow \{N_{SM,1} = 1, C1_{p,1} = C_{o,1}, I1_{p,1} = I_{o,1}\}$$

$$C_{o,2} \neq 0 \rightarrow \{N_{SM,1} = N_{SM,1} + 1, \begin{cases} N_{SM,1} = 1 \rightarrow \{C1_{p,1} = C_{o,2}, I1_{p,1} = I_{o,2}\} \\ N_{SM,1} = 2 \rightarrow \{C2_{p,1} = C_{o,2}, I2_{p,1} = I_{o,2}\} \end{cases}\}$$

4. Последовательно формируется описание результата:

- 4.1. При $N_{SM,1} = 0$ результат – число

$$N_{SM,1} = 0 \rightarrow \{n_s = n_{p,1}, C_s = 0, I_s = 0\}, \text{return};$$

- 4.2. При $N_{SM,1} = 1$ результат – «старая», то есть ранее полученная, символьная переменная или произведение численного сомножителя и символьной переменной

$$N_{SM,1} = 1 \rightarrow \begin{cases} |n_{p,1}| = 1 \rightarrow \{n_s = n_{p,1}, C_s = C1_{p,1}, I_s = I1_{p,1}\}, \text{return} \leftarrow \text{"старая" СИ} \\ n_{p,1} \neq 1 \rightarrow \{n_{PDT,l_p} = n_{p,1}, C1_{PDT,l_p} = C1_{p,1}, I1_{PDT,l_p} = I1_{p,1}\} \\ \{C_{SDT,l_s} = C_{IVA}, I_{SDT,l_s} = l_s, PB_{SDT,l_s} = l_p, PE_{SDT,l_s} = l_p\} \leftarrow \text{число} * СИ \\ l_p = l_p + 1 \\ \{n_s = 1, C_s = C_{IVA}, I_s = l_s\} \\ l_s = l_s + 1 \\ \text{return} \end{cases}$$

- 4.3. При $N_{SM,1} = 2$ результатом является произведение двух символьных переменных

$$N_{SM,1} = 2 \rightarrow \begin{cases} \{n_{PDT,l_p} = n_{p,1}, C1_{PDT,l_p} = C1_{p,1}, I1_{PDT,l_p} = I1_{p,1}, C2_{PDT,l_p} = C2_{p,1}, I2_{PDT,l_p} = I2_{p,1}\} \\ \{C_{SDT,l_s} = C_{IVA}, I_{SDT,l_s} = l_s, PB_{SDT,l_s} = l_p, PE_{SDT,l_s} = l_p\} \\ l_p = l_p + 1 \\ \{n_s = 1, C_s = C_{IVA}, I_s = l_s\} \\ l_s = l_s + 1 \end{cases}$$

Здесь целые переменные l_p и l_s обозначают номера первых свободных строк в таблицах PDT и SDT соответственно.

Операции сложения и вычитания скалярных величин, скалярного произведения двух (3×1) –векторов выполняются аналогично, но из-за большого количества подлежащих учету условий и ограниченного объема статьи здесь не приводятся. На основе скалярного произведения векторов реализуются процедуры выполнения векторно-матричных операций. Оптимизирующие формулы программируются с использованием операций только со скалярными операндами.

5. Исключение неиспользуемых переменных и СМВ. Отдельные СП, обозначающие суммы произведений, в последующих СМВ могут умножаться на ноль и не влиять на окончательный результат вычислений. Соответственно они должны быть исключены из формируемой в символьном виде процедуры. Это обеспечивается следующим образом.

При реализации векторно-матричного алгоритма описания элементов целевых матриц (индексируемые СП) заносятся в строки таблицы SDT, в которых устанавливается значение признака использования, равное 1. После завершения этого процесса строки SDT просматриваются в обратной последовательности, от последней к первой. Если в очередной строке признак использования равен 1, то выбираются соответствующие ей строки таблицы PDT. В них индексы символьных сомножителей, являющихся элементами массива промежуточных переменных, указывают на строки SDT, которые также отмечаются как используемые. В результате суммы, на которые отсутствуют ссылки, не рассматриваются при последующей генерации исходного кода вычислительной процедуры в символьном виде. Элементы массива промежуточных переменных, которые обозначают используемые СМВ, нумеруются заново, получают значение новых индексов.

В процессе исключения неиспользуемых СМВ подсчитывается размерность массива промежуточных переменных, число выполняемых скалярных математических операций. Например, для расчета вектора в правой части уравнений динамики 7-степенного космического манипулятора типа SSRMS необходимо использовать 365 скалярных промежуточных переменных и выполнить 281 умножение и 219 сложений. Число промежуточных переменных невелико, они могут быть описаны не как элементы массива, а как уникальные добавлением индексов к их идентификатору, что позволяет ускорить вычисления.

Заключение. Предложена модель данных для компьютерной оптимизации в символьном виде наиболее эффективных рекуррентных векторно-матричных алгоритмов расчета движения систем многих тел. Для 7-звенного манипулятора она обеспечивает примерно 5-кратное уменьшение объема вычислений по сравнению с численной реализацией. Формируемые в символьном виде процедуры могут быть использованы в системах моделирования движения сложных МС в реальном времени.

Список литературы

1. Грошева М.В., Ефимов Г.Б. О системах аналитических вычислений на ЭВМ // Пакеты прикладных программ: Аналитические преобразования. – М.: Наука, 1988. – С. 5-38.
2. List of computer algebra systems [Электронный ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/List_of_computer_algebra_systems.
3. Martin W.A. Fateman R.J. The MACSYMA system // Proceedings of the 2nd Symposium on Symbolic and Algebraic Manipulation. – Los Angeles, 1971. – P. 59-75.
4. Hearn A.C. REDUCE User's Manual: Version 3.2. – Santa Monica: Ren Corporation, 1985. – 188 p.
5. Maple (software) [Электронный ресурс]. – Режим доступа: [https://en.wikipedia.org/wiki/Maple_\(software\)](https://en.wikipedia.org/wiki/Maple_(software)).
6. Ахо А.В., Хопкрофт Д.Э., Ульман Д.Д., Структуры данных и алгоритмы: Пер. с англ. – СПб.: Изд-во «Диалектика», 2019. – 400 с.
7. Климов Д.М. Методы компьютерной алгебры в задачах механики / Под ред. А.Ю. Ишлинского. – М.: Наука, 1989. – 213 с.
8. Грошева М.В., Ефимов Г.Б., Самсонов В.А. История использования аналитических вычислений в задачах механики. – М.: Институт прикладной математики им. М.В. Келдыша РАН, 2005. – 87 с.
9. Почтаренко М.В. Комплекс программ по анализу стационарных движений механических систем // Пакеты прикладных программ: Методы и разработки. – Новосибирск: Наука, 1981. – С. 82-92.
10. Проворов Л.В., Штаркман В.С. АЛЬКОР – система аналитических вычислений: Часть 2. – М.: Препринт № 166 ИПМ АН СССР, 1982. – 38 с.
11. Банщиков А.В, Бурлакова Л.А., Иванова Г.И., Симонов С.А. Пакет символьных вычислений “МЕХАНИК”: Задачи и структура // Пакеты прикладных программ. Итоги и применения. – Новосибирск: Наука, 1986. – С. 96-105.
12. Городецкий О.М. Специализированная система MMANG для проведения аналитических выкладок в механике сложных систем твердых тел // Пакеты прикладных программ: Аналитические преобразования. – М.: Наука, 1988. – С. 115-128.
13. Вукобратович М., Стокич Д., Кирчански Н. Неадаптивное и адаптивное управление манипуляционными роботами: Пер. с англ. – М.: Изд-во «Мир», 1989. – 376 с.
14. Погорелов Д.Ю. О кодировании символьных выражений при синтезе уравнений движения систем твердых тел // Известия РАН. Техническая кибернетика. – 1993. – №6. – С. 209-213.
15. Computerized Symbolic Manipulation in Mechanic / Edited by E. Kreuzer. – Springer-Verlag Wien. 1994. – 262 p.
16. Степаненко Ю.А. Алгоритм анализа динамики пространственных механизмов с разомкнутой кинематической цепью // Механика машин, выпуск 44. – М.: Наука, 1974. – С. 77-88.
17. Walker M.W., Orin D.E. Efficient Dynamic Computer Simulation of Robotic Mechanisms // Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control. 1982, vol. 104, pp. 205-211.

18. Верещагин А.Ф. Метод моделирования на ЦВМ динамики сложных механизмов роботов-манипуляторов // Известия АН СССР, Техническая кибернетика. – 1974. – №6. – С. 89-94.
19. Armstrong W.W. Recursive solution to the equation of motion of an n-link manipulator // Proceedings of the 5th World Congress on Theory of Machines and Mechanisms. – Montreal, 1979. – P. 1343-1346.
20. Featherstone R. The calculation of robot dynamics using articulated-body inertias // International Journal of Robotic Research. 1983, vol. 2, no. 1, pp. 13-30.
21. Nguen P.K., Hughes P.C. Teleoperation: from the Space Shuttle to the Space Station // Progress in Astronautics and Aeronautics: Teleoperation and Robotics in Space. – Washington: AIAA Inc., 1997, vol. 161, pp. 353-410.

References

1. Grosheva M.V., Efimov G.B. About analytical computation systems for electronic computing machines. // Applied program packages. Analytical transformations. – M.: Science, 1988. – P. 5-38.
2. List of computer algebra systems [Electronic resource]. – Access mode: https://en.wikipedia.org/wiki/List_of_computer_algebra_systems.
3. Martin W.A. Fateman R.J. The MACSYMA system // Proceedings of the 2nd Symposium on Symbolic and Algebraic Manipulation. – Los Angeles, 1971. – P. 59-75.
4. Hearn A.C. REDUCE User's Manual: Version 3.2. – Santa Monica: Ren Corporation, 1985. – 188 p.
5. Maple (software) [Электронный ресурс]. – Режим доступа: [https://en.wikipedia.org/wiki/Maple_\(software\)](https://en.wikipedia.org/wiki/Maple_(software)).
6. Aho A.V., Hopcroft J.E., Ullman J.D. Data structures and algorithms: Transl. from Engl. – SPb.: Publ. house «Dialectics», 2019. – 400 p.
7. Klimov D.M. Computer algebra methods in mechanics problems / Edited by A.Yu. Ishlinsky. – M.: Science, 1989. – 213 p.
8. Grosheva M.V., Efimov G.B., Samsonov V.A. A history of usage of analytical computing in mechanics. – M.: Institute of applied mathematics n.a. M.V. Keldysh RAS, 2005. – 87 p.
9. Pochtarenko M.V. Program complex for analysis of stationary motion of mechanical systems // Packages of applied programs: methods and developments. – Novosibirsk: Science, 1981. – P. 82-92.
10. Provorov L.V., Shtarkman V.C. ALCOR – symbolic computing system: Part 2. – M.: Preprint no. 166 of IPM SU Science academy, 1982. – 38 p.
11. Banshchikov A.V., Burlakova L.A., Ivanova G.I., Simonov S.A. “MECHANIK” – program package for symbolic computing: Tasks and structure. // Packages of applied programs: outcomes and applications. – Novosibirsk: Science, 1986. – P. 96-105.
12. Gorodecky O.M. Specialized system MMANG for analytical computing in mechanics of complex rigid body systems. // Packages of applied programs: analytical transformations. – M.: Science, 1988. – P. 115-128.
13. Vucobratovich M., Stokich D., Kirchasky N. Non-adaptive and adaptive control of manipulation robots: Transl. from Eng. – M.: Publ. house «World», 1989. – 376 p.
14. Pogorelov D.Y. About coding of symbolic expressions by synthesis of motion equations of rigid body systems // Proceedings of RAS. Technical cybernetics. 1993, no 6, pp. 209-213.

15. Computerized Symbolic Manipulation in Mechanic. / Edited by E. Kreuzer. – Springer-Verlag Wien. 1994. – 262 p.
16. Stepanenko Y.A. An algorithm for dynamic analysis of spatial mechanisms with open kinematical chain // Machine mechanics, issue 44. –M.: Science, 1974. – P. 77-88.
17. Walker M.W., Orin D.E. Efficient Dynamic Computer Simulation of Robotic Mechanisms // Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control. 1982, vol. 104, pp. 205-211.
18. Vereshchagin A.F. Computer simulation method of dynamics of complex mechanisms of robot-manipulators // Proceedings of SU Science academy. Technical cybernetics. 1974, no. 6, pp. 89-94.
19. Armstrong W.W. Recursive solution to the equation of motion of an n-link manipulator // Proceedings of the 5th World Congress on Theory of Machines and Mechanisms. – Montreal, 1979. – P. 1343-1346.
20. Featherstone R. The calculation of robot dynamics using articulated-body inertias // International Journal of Robotic Research. 1983, vol. 2, no. 1, pp. 13-30.
21. Nguen P.K., Hughes P.C. Teleoperation: from the Space Shuttle to the Space Station // Progress in Astronautics and Aeronautics: Teleoperation and Robotics in Space. – Washington: AIAA Inc., 1997, vol. 161, pp. 353-410.

Яскевич Андрей Владимирович – доктор физико-математических наук, эксперт по динамике электромеханических систем	Yaskevich Andrey Vladimirovch – doctor of physic-mathematical sciences, expert on electro-mechanical systems dynamics
andrey.yaskevich@rsce.ru	

Received 10.08.2025