ПРОГРАММИРОВАНИЕ, 2013, № 3, с. 47–57

КОМПЬЮТЕРНАЯ АЛГЕБРА

УДК 004.421.6:514.8

ТЕНЗОРНЫЕ РАСЧЕТЫ В СИСТЕМАХ КОМПЬЮТЕРНОЙ АЛГЕБРЫ

© 2013 г. А.В. Королькова, Д.С. Кулябов, Л.А. Севастьянов
Российский университет дружбы народов
117198 Москва, ул. Миклухо-Маклая, д.6
E-mail: sevleonid@yandex.ru
Поступила в редакцию 25.07.2012

В статье рассмотрены три вида тензорных расчётов. В соответствии с ними авторы попытались сформулировать критерии, которым должна удовлетворять система компьютерной алгебры для работы с тензорами. Сделан краткий обзор текущего состояния тензорных вычислений в разных системах компьютерной алгебры. Тензорные расчёты проиллюстрированы соответствующими примерами, реализованными в конкретных системах: Cadabra и Maxima.

1. ВВЕДЕНИЕ

Тензорные вычисления используются во многих областях физики. Следует заметить, что во всей своей мощи формализм тензорного анализа проявляется не во всех областях, достаточно часто используют его упрощённые варианты.

Каждая тензорная операция сама по себе достаточно проста. Однако даже при стандартных вычислениях приходится выполнять множество элементарных операций. Эти операции требуют большой внимательности и скрупулёзности. Именно поэтому в данной области актуальны разные упрощения нотации, оптимизация операций (например, тензорные диаграммы Пенроуза).

Одной из задач систем компьютерной алгебры является освобождение исследователя от рутинных операций, что актуально и в случае тензорного исчисления.

2. ОСНОВНЫЕ ОБЛАСТИ ПРИМЕНЕНИЯ И ТИПЫ ЗАПИСИ ТЕНЗОРОВ

Чтобы определить основные виды операции с тензорами, рассмотрим основные области их применения.

2.1. *Безындексные вычисления для теоретических построений*

Безындексные вычисления обычно применяются в теоретических построениях и часто противопоставляются компонентным вычислениям. Посмотрим, как можно реализовать основные тензорные операции в безындексном случае.

- *Сложение тензоров.* При сложении двух тензоров валентности $\left[\begin{smallmatrix} p \\ q \end{smallmatrix}\right]$ получаем тензор валентности $\left[\begin{smallmatrix} p \\ q \end{smallmatrix}\right]$:

$$A + B = C. \tag{1}$$

  Сложение тензоров задаёт структуру абелевой группы.

- *Тензорное умножение.* При тензорном умножении тензора A с валентностью $\left[\begin{smallmatrix} p \\ q \end{smallmatrix}\right]$ на тензор D с валентностью $\left[\begin{smallmatrix} r \\ s \end{smallmatrix}\right]$ получаем тензор E с валентностью $\left[\begin{smallmatrix} p+r \\ q+s \end{smallmatrix}\right]$:

$$A \otimes D = E. \tag{2}$$

  Тензорное умножение задаёт структуру некоммутативной полугруппы.

- *Операция свёртывания.* Обозначим операцию свёртывания тензоров по последним индексам через $\mathfrak{C}$. Тогда под действием этой операции тензор F с валентностью $\left[\begin{smallmatrix} p+1 \\ q+1 \end{smallmatrix}\right]$ переходит в тензор G с валентностью $\left[\begin{smallmatrix} p \\ q \end{smallmatrix}\right]$:

$$\mathfrak{C}F = G. \tag{3}$$

- *Операция перестановки индексов.* Данная операция необходима для задания симметрии тензоров (например, коммутатора или антикоммутатора тензора), для расширения операции свёртывания на свёртку по произвольным индексам. Однако в рамках безындексного подхода обозначить эту операцию нельзя. Впрочем, простейшие симметрии мы можем явно указать в описании объекта (при этом для однозначности придётся наложить ограничения на валентность).

2.2. *Векторные вычисления*

Векторное исчисление – простейший вариант тензорного исчисления (вектор – тензор валентности один). Вектор a^n размерности N представляется как совокупность набора компонент $n = \overline{1, N}$, зависящего от базиса, и линейного закона преобразования компонент при изменении базиса. Часто используемые операции – построение разнообразных дифференциальных операторов и замена базиса. Наиболее распространённые операторы: градиент, дивергенция, ротор (специфична для трёхмерного пространства $\mathbb{R}^3$) [1, 2]).

Для компонентных расчётов необходимо определить базис, метрику и связность (а соответственно, и ковариантную производную). В векторном исчислении широкое распространение получил голономный базис, который строится как совокупность частных производных от координат в касательном расслоении и дуального базиса как 1-формы в кокасательном расслоении:

$$\vec{\delta}_i = \frac{\partial}{\partial x^i}, \quad \vec{\delta}^i = dx^i. \tag{4}$$

Связность и метрика строятся таким образом, чтобы ковариантная производная от метрики равнялась нулю:

$$\nabla_k g_{ij} = 0. \tag{5}$$

В этом случае связность и метрика согласованы [3].

Следует заметить, что в векторных вычислениях также часто используется специальный неголономный базис, позволяющий не различать контравариантные и ковариантные вектора, сохранять размерность при замене координат[1] (подробнее см. в работе [1]):

$$\vec{\delta}_i = \frac{\partial}{\partial s^i}, \quad \vec{\delta}^i = ds^i, \quad ds^i = h^i_j dx^j. \tag{6}$$

Здесь ds^i – элемент длины по соответствующей координате, h^i_j – коэффициенты неголономности (в случае ортогональных координат – коэффициенты Ламе).

2.3. *Дираковские 4-спиноры*

Специальным случаем тензорных объектов являются спиноры (называемые также спинтензорами). В частности спиноры являются представлениями группы Лоренца с полуцелым старшим весом. Обычные тензоры являются представлениями с целочисленным старшим весом.

По историческим причинам наиболее часто в исследованиях используются дираковские 4-спиноры, которые применяют для записи уравнений Дирака, описывающих фермионы со спином $\frac{1}{2}$. Дираковские 4-спиноры суть неприводимые спиноры для случая $n = 4$ и $s = \pm 2$, где n — размерность векторного пространства, $s = n - 2u$ — его сигнатура, u — число отрицательных значений диагонального метрического тензора g_{ab}.

Обычно для манипуляции с дираковскими спинорами используют γ-матрицы, получаемые из уравнения Клиффорда–Дирака [4]:

$$\gamma_{(a}\gamma_{b)} = g_{ab}\hat{I}, \tag{7}$$

где γ_a — матрицы $N \times N$, g_{ab} — метрический тензор, $\hat{I}$ — единичная матрица $N \times N$, N — размерность спинорного пространства:

$$N = \begin{cases} 2^{n/2}, & \text{чётное } n, \\ 2^{n/2-1/2}, & \text{нечётное } n. \end{cases} \tag{8}$$

γ-матрицы являются элементами алгебры Клиффорда, порождающими линейное преобразование спинового пространства.

[1]При преобразовании длина переходит в длину, угол в угол и т.д.

Поскольку γ-матрицы можно рассматривать как коэффициенты перехода от спинового пространства к векторному, то более строго следует ввести спиновые коэффициенты и записать уравнение (7) следующим образом:

$$\gamma^{\sigma}_{a\rho}\gamma^{\tau}_{b\sigma} + \gamma^{\sigma}_{b\rho}\gamma^{\tau}_{a\sigma} = 2g_{ab}\delta^{\sigma}_{\rho}. \tag{9}$$

Для построения полной алгебры Клиффорда необходимы ещё и произведения γ-матриц, однако в силу (7) достаточно рассматривать только антисимметризованные произведения:

$$\gamma_{ab\ldots d} := \gamma_{[a}\gamma_b \cdots \gamma_{d]}. \tag{10}$$

Также вводится элемент γ_5:

$$\gamma_5 := \frac{i}{4!}e^{abcd}\gamma_a\gamma_b\gamma_c\gamma_d, \tag{11}$$

где e^{abcd} — альтернирующий тензор.

Манипуляции с γ-матрицами сводятся к набору соотношений, следующих из алгебраических симметрий, например

$$\gamma^a\gamma_a = 4\hat{I}, \tag{12}$$

$$\gamma^a\gamma^b\gamma^c\gamma_a = 4g^{bc}\hat{I}, \tag{13}$$

$$\gamma_a\gamma_b = \gamma_{ab} + g_{ab}\hat{I}, \tag{14}$$

$$\gamma_a\gamma_b\gamma_c = \gamma_{abc} + g_{ab}\gamma_c + g_{bc}\gamma_a - g_{ac}\gamma_b. \tag{15}$$

2.4. *Тензорные вычисления в общей теории относительности*

Общая теория относительности стала первой физической теорией, потребовавшей всю мощь дифференциальной геометрии и тензорных вычислений [5]. В вычислениях возникают громоздкие тензорные конструкции, которые можно упрощать, учитывая симметрии тензоров. Обычно выделяют одноэлементные (monoterm) и многоэлементные (multiterm) симметрии. Одним из основных элементов теории является тензор Римана, обладающий как простейшими одноэлементными, так и сложными многоэлементными симметриями типа тождеств Бьянки.

Одноэлементные симметрии соответствуют простым перестановочным симметриям и задаются группой перестановок. Для тензора Римана, например, имеем:

$$R_{bacd} = -R_{abcd}, \quad R_{cdab} = R_{abcd}. \tag{16}$$

Многоэлементные симметрии задаются алгеброй перестановок. Тождество Бьянки имеет вид[2]:

$$R_{a(bcd)} = R_{abcd} + R_{acdb} + R_{adbc} = 0. \tag{17}$$

Дифференциальное (второе) тождество Бьянки имеет вид[3]:

$$R_{ab(cd;e)} = \nabla_e R_{abcd} + \nabla_c R_{abde} + \nabla_d R_{abec} = 0. \tag{18}$$

Симметрии наиболее естественно задавать с помощью диаграмм Юнга [6]. Причём наличие предопределённых классов тензоров не отменяет необходимость в явном задании симметрии. Например, тензор Римана R_{abcd} в разных источниках имеет симметрии $\begin{array}{|c|c|}\hline a & c \\ \hline b & d \\ \hline\end{array}$ или $\begin{array}{|c|c|}\hline a & b \\ \hline c & d \\ \hline\end{array}$.

2.5. *Типы записи тензоров*

Таким образом, опираясь на рассмотренные выше виды тензорных вычислений, можно выделить три типа записи тензоров: компонентная запись, запись с абстрактными индексами и безындексная запись. Каждый тип имеет свою специфику и область применения.

Компонентные индексы, фактически, превращают тензор в набор скалярных величин, применяемых при конкретных расчётах. Обычно оперировать с компонентными индексами есть смысл лишь после упрощения тензорного выражения и учёта всех его симметрий.

Безындексную запись часто используют, если исследователя интересует не конечный результат, а симметрии тензоров. Однако эта форма записи страдает недостатком выразительности: тензор рассматривается как целостный объект, соответственно и симметрии возможно рассматривать лишь те, которые относятся к тензору в целом. Для работы с объектами сложной структуры приходится изобретать новые обозначения либо добавлять словесные пояснения. Эту проблему и должны снять абстрактные индексы [7].

Абстрактные индексы следует рассматривать как усовершенствование безындексной записи тензора. Абстрактный индекс обозначает лишь принадлежность тензора к определённому пространству, а не следование тензорному правилу

[2]Круглые скобки в (17) обозначают симметризацию.

[3]Точка с запятой в (18) означает ковариантную производную.

преобразования (в отличие от компонентных индексов). В этом случае возможно рассмотрение как симметрий, охватывающих весь тензор (все его индексы), так и симметрий отдельных групп индексов.

3. ТЕНЗОРНЫЕ ВЫЧИСЛЕНИЯ И СИСТЕМЫ КОМПЬЮТЕРНОЙ АЛГЕБРЫ

Современные системы компьютерной алгебры способны решать задачи достаточно широкого спектра и из разных областей знаний. Есть системы как узко специализированные, так и с претензией на универсальность (обзоры некоторых систем см., например, в работах [8, 9, 10]). Рассмотрим некоторые системы компьютерной алгебры, в которых в той или иной степени реализована возможность работы с тензорами.

3.1. Требования к системе компьютерной алгебры

Три типа записи тензоров соответствуют трём видам тензорных аналитических вычислений, что приводит к определённым требованиям, предъявляемым системе компьютерной алгебры.

Безындексные вычисления манипулируют с тензорами как с целостными алгебраическими объектами. В данном случае возможно либо задание самого простейшего типа симметрии (объект является представлением какой-либо группы или алгебры), либо использование объектов с заранее заданной симметрией.

Абстрактные индексы требуют возможности задания сложных типов симметрии, например, через диаграммы Юнга. Кроме того, необходимо уметь работать с немыми индексами, задавать и учитывать их при приведении к каноническому виду. Оба вида абстрактных вычислений используют информацию о симметриях для приведения к каноническому виду и упрощения тензорных выражений.

Компонентные индексы требуют по сути скалярной системы компьютерной алгебры, возможно наличия простейших операций с матрицами. Фактически задаётся конкретная система координат и метрика. Поскольку все операции производятся компонентами, теряется информация о тензоре как целостном объекте, об его симметриях. Поэтому все операции с симметриями и приведение к каноническому виду должны быть выполнены на предыдущем этапе исследования.

3.2. Нотация

Использование систем компьютерной алгебры зачастую предполагает интерактивную работу пользователя. В этом случае удобство нотации играет важнейшую роль. Следует заметить, что исторически математическая запись тензоров следует нотации системы TEX, а именно тензор T^a_b записывается как `T^{a}_{b}`. Поэтому использование такой нотации было бы вполне естественным. Такой подход реализован в системе *Cadabra*. Однако *Cadabra* — специализированная система для тензорных вычислений. При введении тензорной нотации в системах компьютерной алгебры общего назначения следует учитывать ограничения этих систем (например, знак `^` обычно зарезервирован и используется для возведения в степень).

Поскольку системы общего назначения работают с функциями, а основной внутренней структурой данных является список, то используется функционально-списочная нотация. В качестве имени функции может задаваться имя тензора, а ковариантные и контравариантные индексы задаются либо префиксом (например, как в *xAct*):

```
T(a,-b),
```

либо позиционно (например, как в *Maxima*):

```
T([a],[b]).
```

Возможно также использование ассоциативных списков, например таким образом:

```
Tensor[Name["T"], Indices[Up[a], Down[b]]].
```

Рассмотрим наиболее интересные для практического применения реализации тензорных вычислений в разных системах компьютерной алгебры.

3.3. Cadabra

Cadabra [`http://cadabra.phi-sci.com/`] относится к типу специализированных систем компьютерной алгебры. Область её специализации — теория поля. Поскольку сложные тензорные расчёты являются неотъемлемой частью теории поля, неудивительно, что поддержка тензорных расчётов находится в этой системе на высоком уровне.

Однако в теории поля оперируют в основном абстрактными индексами, компонентным вычислениям уделяется гораздо меньше внимания. Именно поэтому, наверное, компонентные вычисления пока ещё не реализованы в Cadabra. Хотя данная возможность и стоит в плане на реализацию.

Впрочем, компонентные вычисления требуют от системы компьютерной алгебры функционала системы общего назначения, который отсутствует в системе Cadabra.

3.4. Maxima

Maxima [`http://maxima.sourceforge.net/`] — одна из основных свободных систем компьютерной алгебры общего назначения. Maxima произошла от системы Macsyma, разрабатывавшейся в MIT с 1968 по 1982 годы.

В Maxima реализованы все три типа тензорных вычислений [11]:

- пакет *atensor* — безындексные алгебраические вычисления (в него заложен набор основных алгебр; основное предназначение — упрощение тензорных выражений при помощи манипуляций как с одноэлементными, так и с многоэлементными симметриями);
- пакет *ctensor* — компонентные вычисления (есть возможность манипулирования с метрикой, связностями; в пакет включён набор наиболее употребимых метрик);
- пакет *itensor* — вычисления с использованием (абстрактных) индексов.

Реализация дублирует функциональность пакета Macsyma и фактически не развивается. Возможности данных пакетов вряд ли можно считать удовлетворительными на сегодняшний день.

3.5. Reduce

Reduce [`http://www.reduce-algebra.com/`], [12] — одна из старейших (из ныне живых) систем компьютерной алгебры общего назначения. В 2009 году сменила лицензию с коммерческой на лицензию BSD-типа. Правда, заметим, сделано это было достаточно поздно, сообщество в это время занималось другими свободными системами компьютерной алгебры, и поэтому большого преимущества от перехода к свободной лицензии система не получила.

В основную систему входят:

- пакет *atensor*, упомянутый ранее;
- пакет *redten* [`http://www.scar.utoronto.ca/~harper/redten.html`] предназначен для компонентных вычислений.

3.6. Maple

Система Maple [`http://www.maplesoft.com/products/Maple/index.aspx`] является коммерческой системой компьютерной алгебры общего назначения. Имеет в своём составе также и средства для численных вычислений.

- Система Maple имеет встроенные средства для манипуляции с тензорными компонентами, которые не уступают аналогичным средствам в других системах компьютерной алгебры. Фактически это два пакета:
  - пакет *tensor* изначально был направлен на решение задач общей теории относительности и предназначен для компонентных вычислений;
  - в рамках мощного пакета *DifferentialGeometry* (вошедшего на данный момент в состав основной системы) присутствует подпакет *Tensor*, предназначенный также для компонентных вычислений. Его большим преимуществом является возможность использования не только тензорного анализа, но и всей мощи дифференциальной геометрии (например, использование симметрий групп и алгебр Ли);
- пакет *GRTensor II* [`http://grtensor.phy.queensu.ca/`] (лицензия GPL) является одним из самых мощных пакетов компонентных тензорных вычислений.

3.7. Mathematica

Mathematica [`http://www.wolfram.com/mathematica/`] — коммерческая система компьютерной алгебры общего назначения компании Wolfram Research. Содержит в своём составе целый комплекс расчётных и интерактивных инструментов (для создания математических учебных пособий).

- Пакет MathTensor [`http://smc.vnet.net/MathTensor.html`], [13] — коммерческий пакет, предназначенный в первую очередь для алгебраических манипуляций с тензорами.

- Пакет Cartan (Tensors in Physics) [`http://www.adinfinitum.no/cartan/`] — коммерческий пакет, предназначенный в первую очередь для вычислений в общей теории относительности. Поскольку выполняются вычисления в конкретных метриках, пакет оперирует компонентными индексами.

- Пакет Ricci [`http://www.math.washington.edu/~lee/Ricci/`] в основном поддерживает алгебраические манипуляции, также имеет и некоторые элементы компонентных операций. Находится в состоянии стагнации.

- Набор пакетов *xAct* [`http://www.xact.es/`] (лицензия GPL) охватывает операции как с абстрактными, так и с компонентными индексами.

В следующем разделе рассмотрим подробнее две системы компьютерной алгебры: *Cadabra* и *Maxima*.

Важнейшим критерием выбора этих двух систем стала лицензия — рассматривались только свободные системы компьютерной алгебры. Таким образом были исключены из рассмотрения пакеты расширения к коммерческим системам компьютерной алгебры (вне зависимости от лицензии на сами пакеты).

Наиболее развитая на данный момент свободная система компьютерной алгебры для тензорных манипуляций — *Cadabra*. Однако пока она не поддерживает компонентные вычисления. Поэтому в качестве компаньона к ней выбрана система *Maxima*.

Возможным претендентом на рассмотрение могла стать система *Axiom*, однако на данный момент она распалась на несколько не вполне совместимых ответвлений.

4. ОПЕРАЦИИ НАД ТЕНЗОРАМИ В СИСТЕМЕ CADABRA

В качестве демонстрации возможностей Cadabra рассмотрим выполнение разных операций над тензорами в этой системе. Напомним, что в текущей версии Cadabra отсутствует возможность работы с компонентами, поэтому компонентные операции рассмотрены не будут.

4.1. Безындексные вычисления

Зададим правила коммутации и проверим, что они выполняются:

```
{A,B}::Commuting.
{C,D}::AntiCommuting.
```

- Случай коммутирующих тензоров:

```
B A;
```

$$1 := BA;$$

```
@prodsort!(%);
```

$$1 := AB;$$

- Случай антикоммутирующих тензоров:

```
D C;
```

$$2 := DC;$$

```
@prodsort!(%);
```

$$2 := -CD;$$

4.2. Голономные координаты

Покажем, что в случае согласованных метрики и связности ковариантная производная от метрического тензора равна нулю (см. (5)).

Зададим набор индексов, метрику и частную производную:

```
{a,b,c,d,e,f,g,h,i,j,
  k,l,m,n,o,p,q,r,s,t,u#}::Indices.
g_{a b}::Metric.
\partial_{#}::PartialDerivative.
```

Запишем ковариантную производную через символы Кристоффеля и символы Кристоффеля через метрический тензор, чем собственно и зададим согласование метрики со связностью:

```
\nabla := \partial_{c}{g_{a b}} -
  g_{a d}\Gamma^{d}_{b c} -
  g_{d b}\Gamma^{d}_{a c};
```

$$\nabla := \partial_c g_{ab} - g_{ad}\Gamma^d_{bc} - g_{db}\Gamma^d_{ac};$$

```
Gamma:=\Gamma^{a}_{b c} -> (1/2) g^{a d}
  ( \partial_{b}{g_{d c}} +
  \partial_{c}{g_{b d}} -
  \partial_{d}{g_{b c}} );
```

$$Gamma := \Gamma^a_{bc} \to \frac{1}{2} g^{ad}(\partial_b g_{dc} + \partial_c g_{bd} - \partial_d g_{bc});$$

Подставим выражение символа Кристоффеля через метрический тензор в выражение для ковариантной производной:

```
@substitute!(\nabla)(@(Gamma));
```

$$\nabla := \partial_c g_{ab} - \frac{1}{2} g_{ad} g^{de}(\partial_b g_{ec} + \partial_c g_{be} - \partial_e g_{bc}) - \\ - \frac{1}{2} g_{db} g^{de}(\partial_a g_{ec} + \partial_c g_{ae} - \partial_e g_{ac});$$

и раскроем скобки:

```
@distribute!(%);
```

$$\nabla := \partial_c g_{ab} - \frac{1}{2} g_{ad} g^{de} \partial_b g_{ec} - \frac{1}{2} g_{ad} g^{de} \partial_c g_{be} + \\ + \frac{1}{2} g_{ad} g^{de} \partial_e g_{bc} - \frac{1}{2} g_{db} g^{de} \partial_a g_{ec} - \\ - \frac{1}{2} g_{db} g^{de} \partial_c g_{ae} + \frac{1}{2} g_{db} g^{de} \partial_e g_{ac};$$

Поднимаем и опускаем индексы до тех пор, пока не уберём все метрические тензоры, о чём говорит двойной восклицательный знак:

```
@eliminate_metric!!(%);
```

$$\nabla := \partial_c g_{ab} - \frac{1}{2}\partial_b g_{ac} - \frac{1}{2}\partial_c g_{ba} + \frac{1}{2}\partial_a g_{bc} - \\ - \frac{1}{2}\partial_a g_{bc} - \frac{1}{2}\partial_c g_{ab} + \frac{1}{2}\partial_b g_{ac};$$

Далее приведём выражение к каноническому виду и приведём подобные. В результате получим ноль, как и ожидалось:

```
@canonicalise!(%);
```

$$\nabla := \partial_c g_{ab} - \frac{1}{2}\partial_b g_{ac} - \frac{1}{2}\partial_c g_{ab} + \\ + \frac{1}{2}\partial_a g_{bc} - \frac{1}{2}\partial_a g_{bc} - \frac{1}{2}\partial_c g_{ab} + \frac{1}{2}\partial_b g_{ac};$$

```
@collect_terms!(%);
```

$$\nabla := 0;$$

4.3. γ-матрицы

Cadabra имеет развитые средства для работы с γ-матрицами любой размерности. Для определённости будем использовать γ-матрицы дираковских 4-спиноров.

Для упрощения вычислений зададим набор действий, выполняющихся после каждой операции:

```
::PostDefaultRules( @@prodsort!(%),
  @@eliminate_kr!(%),
  @@canonicalise!(%),
  @@collect_terms!(%) ).
```

Зададим индексы и пробегаемые ими значения:

```
{a,b,c,d,e,f}::Indices(vector).
{a,b,c,d,e,f}::Integer(0..3).
```

Размерность пространства будет использоваться при нахождении следа символа Кронекера.

При задании γ-матриц указывается метрика (см. (7)):

```
\gamma_{#}::GammaMatrix(metric=g).
g_{a b}::Metric.
g_{a}^{b}::KroneckerDelta.
```

Теперь продемонстрируем несколько симметрийных тождеств, которым удовлетворяют γ-матрицы.

- Уравнение Клиффорда–Дирака (7):

```
\gamma_{a} \gamma_{b} +
  \gamma_{b} \gamma_{a};
```

$$1 := \gamma_a \gamma_b + \gamma_b \gamma_a;$$

Алгоритм `@join` преобразует попарные произведения γ-матриц в сумму γ-матриц высших валентностей (см. (10)). Дополнительный аргумент `expand` указывает на то, что учитываются правила антисимметризации для γ-матриц:

```
@join!(%){expand};
```

$$1 := 2\, g_{ab};$$

- Свёртка двух γ-матриц (12):

```
\gamma^{a} \gamma_{a};
```

$$2 := \gamma^a \gamma_a;$$

```
@join!(%){expand};
```

$$2 := 4;$$

- Тождество (13):

```
\gamma^{a} \gamma^{b}
  \gamma^{c} \gamma_{a};
```

$$3 := \gamma^a \gamma^b \gamma^c \gamma_a;$$

```
@join!!(%){expand};
```

$$3 := (\gamma^{ab} + g^{ab})(-\gamma_a{}^c + g_a{}^c);$$

```
@distribute!(%);
```

$$3 := -\gamma^{ba} \gamma^c{}_a - 2\,\gamma^{bc} + g^{bc};$$

```
@join!!(%){expand};
```

$$3 := 4\, g^{bc};$$

- Тождество (14):

```
\gamma_{a} \gamma_{b};
```

$$4 := \gamma_a \gamma_b;$$

```
@join!!(%){expand};
```

$$4 := \gamma_{ab} + g_{ab};$$

- Тождество (15):

```
\gamma_{a} \gamma_{b} \gamma_{c};
```

$$5 := \gamma_a \gamma_b \gamma_c;$$

```
@join!!(%){expand};
```

$$5 := (\gamma_{ab} + g_{ab})\gamma_c;$$

```
@distribute!(%);
```

$$5 := \gamma_{ab}\gamma_c + \gamma_c g_{ab};$$

```
@join!!(%){expand};
```

$$5 := \gamma_{abc} + \gamma_a g_{bc} - \gamma_b g_{ac} + \gamma_c g_{ab};$$

- Более сложное тождество:

$$\gamma_{ab}\gamma_{bc}\gamma_{de}\gamma_{ea} = -4\,\gamma_{cd} + 21\, g_{cd}\hat{I} \tag{19}$$

в Cadabra будет иметь вид:

```
\gamma_{a b} \gamma_{b c}
  \gamma_{d e} \gamma_{e a};
```

$$6 := -\gamma_{ab}\gamma_{ca}\gamma_{de}\gamma_{be};$$

```
@join!!(%){expand};
```

$$6 := -(2\,\gamma_{bc} + 3\, g_{bc})(2\,\gamma_{bd} - 3\, g_{bd});$$

```
@distribute!(%);
```

$$6 := -4\,\gamma_{cb}\gamma_{db} - 12\,\gamma_{cd} + 9\, g_{cd};$$

```
@join!(%){expand};
```

$$6 := -4\,\gamma_{cd} + 21\, g_{cd};$$

4.4. *Одноэлементные симметрии*

В качестве примера одноэлементной симметрии рассмотрим симметрии тензора Римана (16). Для этого зададим вначале симметрийные свойства с помощью диаграммы Юнга:

```
R_{a b c d}::TableauSymmetry(shape={2,2},
  indices={0,2,1,3}).
```

В данном примере симметрия имеет вид $\begin{array}{|c|c|}\hline a & c \\ \hline b & d \\ \hline\end{array}$.

Далее произведём симметричную и антисимметричную перестановку индексов. Алгоритм `@canonicalise` приводит операнд к канонической форме, учитывая одноэлементные симметрии.

```
R_{c d a b};
```

$$1 := R_{cdab};$$

```
@canonicalise!(%);
```

$$1 := R_{abcd};$$

```
R_{a b c d} + R_{b a c d};
```

$$2 := R_{abcd} + R_{bacd};$$

```
@canonicalise!(%);
```

$$2 := R_{abcd} - R_{abcd};$$

```
@collect_terms!(%);
```

$$2 := 0;$$

4.5. *Многоэлементные симметрии*

Работу с многоэлементными симметриями продемонстрируем на примере тензора Римана.

Введём обозначения для индексов и производной:

```
{a,b,c,d,e,f,g#}::Indices(vector).
\nabla{#}::Derivative.
```

Симметрию можно задавать с помощью диаграммы Юнга $\begin{array}{|c|c|c|}\hline a & c & e \\ \hline b & d \\ \cline{1-2}\end{array}$, как в предыдущем случае:

```
\nabla_{e}{R_{a b c d}}::TableauSymmetry(
  shape={3,2}, indices={1,3,0,2,4} ).
```

Однако удобнее использовать следующую нотацию:

```
R_{a b c d}::RiemannTensor.
```

Аналогично поступим и с ковариантной производной от тензора Римана, удовлетворяющей дифференциальному тождеству Бьянки:

```
\nabla_{e}{R_{a b c d}}::SatisfiesBianchi.
```

Проверим первое тождество Бьянки (17).

```
R_{a b c d} + R_{a c d b} + R_{a d b c};
```

$$1 := R_{abcd} + R_{acdb} + R_{adbc};$$

```
@young_project_tensor!2(%){ModuloMonoterm}:

@collect_terms!(%);
```

$$1 := 0;$$

Теперь продемонстрируем выполнение второго (дифференциального) тождества Бьянки (18):

```
\nabla_{e}{R_{a b c d}} +
  \nabla_{c}{R_{a b d e}} +
  \nabla_{d}{R_{a b e c}};
```

$$2 := \nabla_e R_{abcd} + \nabla_c R_{abde} + \nabla_d R_{abec};$$

```
@young_project_tensor!2(%){ModuloMonoterm}:

@collect_terms!(%);
```

$$2 := 0;$$

5. ПРИМЕР ТЕНЗОРНЫХ ВЫЧИСЛЕНИЙ В MAXIMA

Поскольку Cadabra на данный момент не поддерживает компонентные вычисления, продемонстрируем их в системе Maxima. В качестве примера рассмотрим запись уравнений Максвелла в цилиндрических координатах в голономном базисе [1].

Загрузим вначале написанный нами небольшой пакет, содержащий определения для дифференциальных операторов:

```
(%i1)  load("diffop.mac")$
```

Зададим цилиндрическую систему координат:

```
(%i2)  ct_coordsys(polarcylindrical)$
```

Посмотрим компоненты метрического тензора g_{ij}:

```
(%i3)  lg;
```

$$(\%o3) \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & r^2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Вычислим и посмотрим компоненты метрического тензора g^{ij}:

```
(%i4)  cmetric()$
```

```
(%i5)  ug;
```

$$(\%o5)\quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{r^2} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Зададим необходимые векторы и определим их зависимость от координат. Из-за ограничений Maxima обозначим j^1 через `j1`, а j_1 через `j_1`:

```
(%i6)  j:[j1,j2,j3]$
       depends(j,cons(t,ct_coords))$
(%i8)  B:[B1,B2,B3]$
       depends(B,cons(t,ct_coords))$
(%i10) D:[D1,D2,D3]$
       depends(D,cons(t,ct_coords))$
(%i12) H:[H_1,H_2,H_3]$
       depends(H,cons(t,ct_coords))$
(%i14) E:[E_1,E_2,E_3]$
       depends(E,cons(t,ct_coords))$
```

Теперь вычислим все части уравнений Максвелла в цилиндрических координатах:

```
(%i16) Div(B);
```

$$(\%o16)\quad \frac{d}{d\,z}\,B3+\frac{d}{d\,\theta}\,B2+\frac{d}{d\,r}\,B1+\frac{B1}{r}$$

```
(%i17) Div(D) -4*%pi*rho;
```

$$(\%o17)\quad \frac{d}{d\,z}\,D3+\frac{d}{d\,\theta}\,D2+\frac{d}{d\,r}\,D1+\frac{D1}{r}-4\,\pi\,\rho$$

```
(%i18) Rot(H)
         + diff(transpose(matrix(D)),t)/c
         - 4*%pi/c*transpose(matrix(j));
```

$$(\%o18)\quad \begin{pmatrix} \frac{\frac{d}{d\,\theta}\,H\_3-\frac{d}{d\,z}\,H\_2}{|r|}+\frac{\frac{d}{d\,t}\,D1}{c}-\frac{4\,\pi\,j1}{c} \\ -\frac{\frac{d}{d\,r}\,H\_3-\frac{d}{d\,z}\,H\_1}{|r|}+\frac{\frac{d}{d\,t}\,D2}{c}-\frac{4\,\pi\,j2}{c} \\ \frac{\frac{d}{d\,r}\,H\_2-\frac{d}{d\,\theta}\,H\_1}{|r|}+\frac{\frac{d}{d\,t}\,D3}{c}-\frac{4\,\pi\,j3}{c} \end{pmatrix}$$

```
(%i19) Rot(E)
         + diff(transpose(matrix(B)),t)/c;
```

$$(\%o19)\quad \begin{pmatrix} \frac{\frac{d}{d\,\theta}\,E\_3-\frac{d}{d\,z}\,E\_2}{|r|}+\frac{\frac{d}{d\,t}\,B1}{c} \\ \frac{\frac{d}{d\,t}\,B2}{c}-\frac{\frac{d}{d\,r}\,E\_3-\frac{d}{d\,z}\,E\_1}{|r|} \\ \frac{\frac{d}{d\,r}\,E\_2-\frac{d}{d\,\theta}\,E\_1}{|r|}+\frac{\frac{d}{d\,t}\,B3}{c} \end{pmatrix}$$

Таким образом, результат совпал с полученными в [1] аналитическими выражениями.

6. ЗАКЛЮЧЕНИЕ

Статья вызвана к жизни тем, что авторы хотели провести громоздкие тензорные вычисления в системе компьютерной алгебры. Поиск соответствующей системы заставил сформулировать набор критериев, которым должна удовлетворять подобная система компьютерной алгебры.

Из-за наличия нескольких типов тензорных расчётов полная реализация тензорных вычислений в системах компьютерной алгебры требует широкого набора возможностей. К сожалению, на данный момент фактически отсутствуют системы с полностью удовлетворительной поддержкой тензоров.

Компонентные тензорные вычисления практически не требуют дополнительных возможностей от универсальной системы компьютерной алгебры. Поэтому пакеты, реализующие данный функционал, представлены наиболее широко (например, *Maxima*, *Maple*, *Mathematica*).

Тензорная нотация сильно отличается от привычной функциональной нотации, которую использует подавляющее большинство систем компьютерной алгебры. Поэтому для эффективной работы с тензорами представляется необходимым иметь специализированную систему (либо специализированную надстройку над универсальной системой), поддерживающую естественную тензорную нотацию (например, *Cadabra*).

В результате не удалось найти систему, полностью удовлетворяющую потребностям тензорного исчисления. Авторы пока остановились на наборе из специализированной системы *Cadabra* и универсальной системы компьютерной алгебры (на данный момент *Maxima*).

СПИСОК ЛИТЕРАТУРЫ

1. *Kulyabov D.S., Korolkova A.V., Korolkov V.I.* Maxwell's Equations in Arbitrary Coordinate System // Bulletin of PFUR. Series "Mathematics. Information Sciences. Physics". 2012. № 1. P. 96–106.
2. *Кулябов Д.С., Немчанинова Н.А.* Уравнения Максвелла в криволинейных координатах // Вестник РУДН. Серия «Математика. Информатика. Физика». 2011. № 2. С. 172–179.
3. *Сарданашвили Г.А.* Современные методы теории поля. Т. 2. Геометрия и классическая механика. М.: УРСС, 1998.
4. *Cartan E.* The Theory of Spinors. Paris: Hermann, 1966.
5. Гердт В.П., Тарасов О.В., Ширков Д.В. Аналитические вычисления на ЭВМ в приложении к физике и математике // Успехи физических наук. 1980. Т. 130. С. 113–147. http://ufn.ru/ru/articles/1980/1/d/
6. *Барут А., Рончка Р.* Теория представлений групп и ее приложения. М.: Мир, 1980.

7. *Penrose R., Rindler W.* Spinors and Space-Time. Two-Spinor Calculus and Relativistic Fields. Camgridge University Press. 1987. V. 1. 472 p.

8. Computer Algebra Information Network. http://www.computeralgebra.nl/systemsoverview//special/systems.html

9. *Кулябов Д. С., Кокотчикова М. Г.* Аналитический обзор систем символьных вычислений // Вестник РУДН. Серия «Математика. Информатика. Физика». 2007. № 1–2. С. 38–45.

10. *Sevastianov L.A., Kulyabov D.S., Kokotchikova M.G.* An Application of Computer Algebra System Cadabra to Scientific Problems of Physics // Письма в ЭЧАЯ. 2009. Т. 6. № 7 (156). С. 39–44.

11. *Toth V.* Tensor Manipulation in GPL Maxima. 2005. arXiv : cs/0503073v2

12. *Gerdt V.P., Tiller P.* A Reduce program for symbolic computation of Puiseux expansions. Dubna: Joint Inst. Nuclear Res., 1991.

13. *Parker L., Christensen S. M.* MathTensor: a system for doing tensor analysis by computer. Addison-Wesley, 1994. 379 p.