



ГОСУДАРСТВЕННЫЙ НАУЧНЫЙ ЦЕНТР РОССИЙСКОЙ ФЕДЕРАЦИИ
ИНСТИТУТ ФИЗИКИ ВЫСОКИХ ЭНЕРГИЙ

ИФВЭ 2004–24
ОАФ, ОМВТ

В.В. Ежела, В.Н. Ларин

Пакет StandardPhysicalConstants 2.1.
(Руководство для пользователя)

Протвино 2004

Аннотация

Ежела В.В., Ларин В.Н. Пакет `StandardPhysicalConstants 2.1`. (Руководство для пользователя): Препринт ИФВЭ 2004–24. – Протвино, 2004. – 26 с.

Представлено детальное описание пакета `StandardPhysicalConstants` версии 2.1 для вычислений средних значений и (ко)вариаций функций от фундаментальных физических постоянных (ФФП) в системе *Mathematica*. Пакет ориентирован на работу с объединенной выборкой (113 постоянных) из оцененных и рекомендованных CODATA(2002) и PDG(2004) значений ФФП с учетом оцененных и рекомендованных значений корреляций их неопределенностей.

Abstract

Ezhela V.V., Larin V.N. `StandardPhysicalConstants 2.1` (Package. User's Guide): IHEP Preprint 2004–24. – Protvino, 2004. – p. 26.

The detailed user's guide of the `StandardPhysicalConstants 2.1` package for *Mathematica* system is given. The package is to calculate average values and (co)variances of the functions dependent upon fundamental physical constants (FPC). The package set of FPC (113 constants) is combined from adjusted and recommended CODATA(2002) and PDG(2004) values, their variances, and correlations.

Содержание

1. Введение	2
2. Структура пакета и его инициализация	2
2.1. Структура пакета.	2
2.2. Инициализация пакета.	3
3. База данных	3
3.1. Структура данных.	3
3.2. Источники и качество данных	4
4. Возможности пакета	5
4.1. Функции управления данными.	5
4.1.1. NewListFPC.	5
4.1.2. PrintDB	6
4.1.3. OutputDB	7
4.1.4. InsertFPC	7
4.1.5. DeleteFPC	9
4.1.6. Copy.	10
4.1.7. Shift	11
4.2. Функции доступа к данным	11
4.2.1. Access.	11
4.2.2. PairCC	12
4.2.3. GroupCC	13
4.2.4. MakeCM	14
4.3. Функции–“калькуляторы”	14
4.3.1. ErrorPropagator.	15
4.3.2. MatrixCharacteristics.	18
4.3.3. Covariator	19
5. Заключение	22
6. Приложение	24

1. Введение

Результаты двух последних последовательных переоценок и согласований системы фундаментальных физических постоянных (ФФП) в 1998 г. [1] и в 2002 г. [2], проведенных в NIST, показали, что получаемые матрицы ковариаций и корреляций — это плохо обусловленные матрицы. Использование плохо обусловленных матриц требует особой аккуратности в высокоточных вычислениях. В частности, игнорирование корреляций погрешностей или их необоснованное округление может приводить к абсурдным результатам (см. пример в работе [3]).

Однако игнорирование корреляций погрешностей ФФП оказывается устоявшейся практикой в вычислениях с использованием ФФП. Во всех популярных системах для научных вычислений [4], [5], [6] используются выборки данных из системы ФФП, рекомендованных CODATA, но без соответствующих корреляторов и нет даже упоминаний о наличии больших (> 90%) коэффициентов корреляций погрешностей ФФП.

Предлагаемый пакет `StandardPhysicalConstants` [3, 7] призван решить проблему учета корреляций погрешностей, рекомендованных CODATA ФФП, в вычислениях физических величин в системе *Mathematica* [8] для задач физики частиц и смежных разделов.

Сразу заметим, что пакет находится в стадии развития, поэтому не все возможности предоставления “комфортного” использования пакета реализованы в настоящей версии. Однако пакет уже можно использовать для проведения корректных вычислений с использованием максимально полной доступной информации о ФФП.

2. Структура пакета и его инициализация

2.1. Структура пакета

Дистрибутив пакета состоит из двух основных файлов (`name.m`), написанных на языке *Mathematica*, и трех “записных книжек” (`notebook — name.nb`):

`Dbfpc113.m` — файл базы данных пакета, содержащий информацию о 113 ФФП.

`Manager.m` — пакет функций управления для доступа, хранения и сопровождения данных.

Он также включает модули для вычисления средних значений, (ко)вариаций и корреляций функций, зависящих от ФФП и других случайных величин, которые могут храниться в рабочей базе данных.

`FPalette.nb` — палитра функций пакета.

`CPalette_113.nb` — палитра пакетных имен постоянных.

`Read_me.nb` — краткое описание пакета.

2.2. Инициализация пакета

Для инициализации пакета необходимо загрузить в текущей сессии *Mathematica* два основных файла с помощью функции `Get[file_name]` или `<<file_name`, где *file_name* — полное имя файла. Для того чтобы использовать “короткие” имена файлов, достаточно поместить их в директорию *Autoload* (создается при инсталляции системы *Mathematica*), или до загрузки файлов указать полный путь к ним с помощью команды `SetDirectory[“full_path”]`. Если любое из двух последних условий выполнено, команды загрузки основных файлов пакета выглядят так (порядок загрузки не имеет значения):

```
In[1] := << Dbfpc113.m
```

```
In[2] := << Manager.m
```

Ввод этих команд может быть также осуществлен с палитры `FPalette.nb`.

3. База данных

Текущая версия базы данных `Dbfpc113.m` содержит информацию о средних значениях, стандартных отклонениях и *известных* коэффициентах корреляции для 113 физических постоянных. В частности, она содержит информацию о всех универсальных и часто используемых константах, таких как скорость света в вакууме (c_0), магнитная и электрическая постоянные (μ_0 и ϵ_0), элементарный заряд (e), постоянная Планка (h) и т.д., а также о наиболее часто используемых электромагнитных, атомных и ядерных постоянных, таких как магнетон Бора (μ_B), постоянная тонкой структуры (α), массы, их отношения, магнитные моменты некоторых элементарных частиц (m_e , m_e/m_μ , μ_p и т.д.). Кроме того, в базу данных включен ряд астрофизических постоянных: гравитационная постоянная (G), планковская масса ($\sqrt{\hbar c_0/G}$), постоянная Хаббла (H_0) и другие. Полный список физических постоянных, включенных в текущую версию базы данных, приведен в **Приложении**.

3.1. Структура данных

Все данные по физическим постоянным объединены в один список с именем `dataBase`, элементами которого являются три списка-таблицы:

$$dataBase = \{ basicTable, standardNames, ccTable \},$$

где *basicTable* — список физических постоянных с элементами-списками вида

$$\{ integer_number, package_name, constant_symbol, value, units \},$$

которые включают порядковый номер константы в этой таблице (ее локальный идентификатор в используемой базе данных — *код*), ее “пакетное” имя (глобальный идентификатор в любой базе данных — основной или рабочей), символьное обозначение, значение (средняя величина и стандартное отклонение в форме “NIST Concise form”)

$$average(uncertainty) \times 10^{scale},$$

и единицы измерения;

standardNames — список “стандартных” имен постоянных (как они определены в источниках данных);

ccTable — таблица коэффициентов корреляции, запасенная как список наддиагональных элементов матрицы корреляции ($r_{i<j}$) с элементами-списками вида

$$\{code_1, code_2, constant_symbol_1, constant_symbol_2, r_{code_1, code_2}\}.$$

Все элементы таблиц, за исключением *кодов* постоянных (целые числа), представляются в текстовом формате (*character string*). Структуры этих таблиц представлены в примерах следующих разделов.

3.2. Источники и качество данных

При формировании базы данных *Dbfpc113.m* использовались два дополняющих друг друга источника [2, 9]. Следует отметить, что источники не синхронизированы по датам проведения переоценок и не согласованы по полноте представления оцененных данных.

Так, переоценка и согласование традиционного набора констант CODATA проводились эпизодически с 1950-х годов, но, начиная с 1998 г., они проводятся каждые 4 года (см. [1], стр. 355). Последняя переоценка и согласование ФФП CODATA произведены по экспериментальным данным и теоретическим формулам, опубликованным до 2003 г. (см. [2], где для большинства рекомендованных CODATA ФФП приведены средние значения, стандартные отклонения и коэффициенты корреляции отклонений).

К сожалению, не все ФФП физики частиц и астрофизики включены в стандартную процедуру переоценки и согласования CODATA. В наборе ФФП CODATA-2002 нет данных по массам W^\pm и Z^0 бозонов, постоянной сильного взаимодействия $\alpha_s(M_Z)$, а оценки постоянных *weak mixing angle*, *Fermi coupling constant*, *tau mass [MeV]*, включенные в рекомендуемый CODATA набор ФФП [2], извлечены из данных, рекомендованных сотрудничеством PDG в 2002 г. [10] и уже частично устаревших.

Отметим, что сотрудничество PDG проводит переоценку ФФП физики частиц и астрофизики (дополнительных к набору ФФП, оцениваемых CODATA) ежегодно с публикацией отчетов о переоценке каждые два года. Последняя переоценка проведена по данным, опубликованным до 2004 г. (см. обзор RPP-2004 [9], в котором также частично приведены данные по ФФП (контролируемым CODATA) из [2], но без соответствующих коэффициентов корреляции отклонений)¹.

Таким образом, во многих приложениях в физике частиц и астрофизике оба источника необходимы. Для исключения использования “несинхронизированных оценок” в пакете *StandardPhysicalConstants* оценки всех ФФП, контролируемых сотрудничеством PDG, взяты из RPP-2004. Отметим также, что всем неизвестным коэффициентам корреляции в нашей базе данных приписано специальное значение "Null".

¹ В наборе CODATA-2002 для многих пар ФФП корреляции отклонений весьма велики, например, $r(m_\mu [MeV], m_e [MeV]) = 0.956$, $r(m_\mu [kg], m_e [kg]) = 0.988$.

4. Возможности пакета

Модули пакета управления данными **Manager.m** обеспечивают выполнение функций, с помощью которых можно манипулировать базой данных (функции управления данными), извлекать информацию о постоянных из базы данных (функции доступа к данным) и выполнять вычисление наблюдаемых или “вторичных” физических постоянных с использованием информации, хранящейся в базе данных. Результатом являются средние значения искомым величин, их стандартные отклонения и ковариации (функции-“калькуляторы”).

4.1. Функции управления данными

Функции этой группы позволяют создавать новые базы данных, добавлять в них новые константы, удалять константы, копировать постоянные из одной базы данных в другую, перемещать константы в пределах одной базы данных, представлять базу данных и отдельные ее элементы (таблицы) в табулированном виде на экране дисплея, а также выводить вновь созданную или модифицированную базу данных в файл.

Список этих функций приведен в следующей таблице.

NewListFPC	создает новую базу данных, используя информацию из уже существующей базы данных.
PrintDB	представляет указанную базу данных или ее элемент (таблицу) в табулированном формате.
OutputDB	выводит указанную базу данных в файл со стандартным именем Dbfpc_n.m .
InsertFPC	вставляет новую постоянную в указанную базу данных.
DeleteFPC	удаляет указанную постоянную или несколько постоянных из базы данных.
Copy	копирует всю информацию для указанных постоянных из базы-источника в базу-получатель.
Shift	изменяет позицию указанной постоянной в базе данных.

4.1.1. NewListFPC

Эта функция создает новую базу данных, используя данные из указанной в качестве параметра базы данных (основной **dataBase** или ранее уже созданной). Такая процедура полезна, если в решаемой пользователем задаче используется небольшое количество физических постоянных, а предполагаемые вычисления достаточно громоздки. Кроме того, во временную (рабочую) базу данных можно включать новые физические величины, отсутствующие в основной базе, или модифицировать информацию для уже существующих постоянных, не изменяя содержания основной базы данных (впрочем, последнее также допускается, так как исходная информация хранится в файле **Dbfpc113.m**).

Функция имеет следующий формат:

```
NewListFPC[ constant_list, data_base ],
```

где *constant_list* — список констант, включаемых в создаваемую базу данных, информация о которых хранится в базе *data_base*; причем постоянные в списке констант могут быть представлены как их кодом в *data_base*, так и пакетным именем:

```
In[3] := wdb=NewListFPC[ {102, 38, "planckConstant"}, dataBase ]
```

```
Out[3] = {{{1, fineStructureConstant,  $\alpha$ , 7.297 352 568(24) \times 10^{-3}, },
           {2, planckConstant, h, 6.626 0693(11) \times 10^{-34}, J s},
           {3, speedOfLightInVacuum, c0, 299 792 458, m s-1}},
          {fine-structure constant, Planck constant, speed of light in vacuum},
          {{1, 2,  $\alpha$ , h, 0.010}, {1, 3,  $\alpha$ , c0, 0.000}, {2, 3, h, c0, 0.000}}}
```

Заметим, что константы в новой базе данных с именем **wdb** упорядочены по алфавиту, хотя во входном списке они расположены в произвольном порядке. Кроме того, константам в этой базе данных присвоены новые коды: 1, 2 и 3 (в основной базе их коды 38, 74 и 102 соответственно). Напомним, что код является *локальным* идентификатором постоянной, зависящим от используемой базы данных, а ее пакетное имя — *глобальным* идентификатором. Структура этой базы данных полностью аналогична структуре основной базы **dataBase**, даже если список констант пуст:

```
In[4] := NewListFPC[ { }, dataBase ]
```

```
Out[4] = {{{Null, Null, Null, Null, Null}}, { }, {{{Null, Null, Null, Null, Null}}}
```

4.1.2. PrintDB

Эта функция позволяет представить указанную базу данных или любую ее таблицу в табулированной форме. Формат функции:

```
PrintDB[ data_base ],
PrintDB[ element_of_data_base ],
```

где *data_base* — имя базы данных, а *element_of_data_base* — один из трех элементов базы данных *data_base[[i]]* ($i = 1, 2, 3$) или имя таблицы, ассоциированное с ним, например, для таблицы коэффициентов корреляции *cstab = data_base[[3]]*. Применение этой функции к сформированной выше базе данных дает следующий результат:

```
In[5] := PrintDB[ wdb ]
```

Constant Table

Code	Name	Symbol	Value	Units
1	fineStructureConstant	α	$7.297\ 352\ 568(24) \times 10^{-3}$	
2	planckConstant	h	$6.626\ 0693(11) \times 10^{-34}$	J s
3	speedOfLightInVacuum	c ₀	299 792 458	m s ⁻¹

Constant Standard Names

1	fine-structure constant
2	Planck constant
3	speed of light in vacuum

Correlation Coefficient Table

i	j	x_i	x_j	r_{ij}
1	2	α	h	0.010
1	3	α	c_0	0.000
2	3	h	c_0	0.000

В то же время, если в качестве параметра функции ввести элемент базы данных, например первый, то выводится только первая таблица.

```
In[6] := PrintDB[ wdb[[1]] ]
```

Constant Table

Code	Name	Symbol	Value	Units
1	fineStructureConstant	α	$7.297\ 352\ 568(24) \times 10^{-3}$	
2	planckConstant	h	$6.626\ 0693(11) \times 10^{-34}$	J s
3	speedOfLightInVacuum	c_0	299 792 458	m s ⁻¹

Аналогичный результат будет получен и для двух других элементов (таблиц).

4.1.3. OutputDB

С помощью этой функции можно вывести вновь созданную базу данных в файл со специальным именем `Dbfpc_n.m` в текущей директории. Этот файл организован точно так же, как основной (`Dbfpc113.m`), включая имена базы данных и ее элементов: $dataBase = \{basicTable, standardNames, ccTable\}$. Его можно переименовать, редактировать, не нарушая структуру базы данных, и использовать вместо основного, загрузив его так, как описано в разделе 2.2. Функция имеет следующий формат:

`OutputDB[data_base],`

где `data_base` — имя новой базы данных, созданной, например, функцией `NewListFPC`.

4.1.4. InsertFPC

Эта функция позволяет добавлять новые (т.е. отсутствующие в основной и рабочих базах) величины в базу данных, указанную в качестве ее параметра. Она имеет следующий формат:

```

InsertFPC[ const_data_list, data_base ],
InsertFPC[ const_data_list, data_base, initcc ],

```

где *const_data_list* — список данных, характеризующих добавляемую физическую величину, физическую или математическую константу, *data_base* — имя модифицируемой базы данных и *initcc* — исходное “значение” коэффициентов корреляции вводимой величины со всеми другими постоянными в базе данных *data_base* (по умолчанию всем им приписывается значение “пробел”, т.е. “ ”).

В дальнейшем исходные “значения” корреляций можно заменить на известные численные значения, представляя их в формате *character string* с помощью штатных средств системы *Mathematica*, или прямым редактированием спасенной в файл (командой **OutputDB**) модифицированной базы данных. Если коэффициенты корреляции для новой величины неизвестны, следует приписать им значение "Null". Если добавляемая величина является *точной*, ее корреляциям следует приписать значение "0.000".

Первый параметр функции (*const_data_list*) представляет собой список из пяти или шести элементов:

$$\{code, "packagename", "symbol", "value", "units", "standardname"\},$$

где все элементы, кроме первого (код величины — целое число), представляются в формате *character string*. Первый и последний элементы являются необязательными. Если опущен код (*code*), новая величина вставляется в базу данных в алфавитном порядке (по пакетному имени “*packagename*”), не нарушая ее упорядоченность. Если опущен последний элемент, в таблицу стандартных имен будет внесено пакетное имя величины (“*packagename*”).

В качестве примера рассмотрим включение в рабочую базу данных **wdb** константы $m_p c^2$ — энергетического эквивалента массы протона в *MeV* (отсутствует в основной базе **Dbfpc113.m**). Необходимые данные возьмем на сайте NIST [2].

```

In[7] := mydb = InsertFPC[ { , "protonMassInMeV", "m_p c_0^2", "938.272 029(80)",
                          "MeV", "proton mass energy equivalent in MeV"}, wdb ];

```

Переопределив в этом примере имя рабочей базы данных, получим новую базу **mydb**, а “старую” (**wdb**) сохраним неизменной. Полученная база данных имеет вид

```

In[8] := PrintDB[ mydb ]

```

Constant Table

Code	Name	Symbol	Value	Units
1	fineStructureConstant	α	$7.297\ 352\ 568(24) \times 10^{-3}$	
2	planckConstant	h	$6.626\ 0693(11) \times 10^{-34}$	J s
3	protonMassInMeV	$m_p c_0^2$	938.272 029(80)	MeV
4	speedOfLightInVacuum	c_0	299 792 458	m s ⁻¹

Constant Standard Names

1	fine-structure constant
2	Planck constant
3	proton mass energy equivalent in MeV
4	speed of light in vacuum

Correlation Coefficient Table

i	j	x_i	x_j	r_{ij}
1	2	α	h	0.010
1	3	α	$m_p c_0^2$	
1	4	α	c_0	0.000
2	3	h	$m_p c_0^2$	
2	4	h	c_0	0.000
3	4	$m_p c_0^2$	c_0	

Теперь остается заполнить “пустые” поля значений коэффициентов корреляции для новой постоянной, и расширение базы данных будет завершено. Выполнив соответствующие операции присваивания

```
In[9] := mydb[[3,2,5]] = "-0.086";  
        mydb[[3,4,5]] = "0.995";  
        mydb[[3,6,5]] = "0.000";
```

получим полную таблицу корреляций:

```
In[10] := PrintDB[ mydb[[3]] ]
```

Correlation Coefficient Table

i	j	x_i	x_j	r_{ij}
1	2	α	h	0.010
1	3	α	$m_p c_0^2$	-0.086
1	4	α	c_0	0.000
2	3	h	$m_p c_0^2$	0.995
2	4	h	c_0	0.000
3	4	$m_p c_0^2$	c_0	0.000

4.1.5. DeleteFPC

Обратная предыдущей операция по удалению постоянной или нескольких постоянных из базы данных осуществляется с помощью функции DeleteFPC, имеющей формат

`DeleteFPC[const, data_base],`

где *const* может быть как идентификатором одной удаляемой постоянной, так и списком идентификаторов удаляемых постоянных; *data_base* — имя базы данных, из которой эти константы удаляются. Заметим, что удаление константы из базы данных означает удаление всех данных, связанных с ней.

4.1.6. Copy

Если в рабочую базу данных необходимо добавить одну или несколько констант, которые включены в другую базу (например, в основную), удобно воспользоваться функцией *Copy*, которая копирует всю информацию для указанных постоянных, включая корреляционные коэффициенты, из базы-источника в базу-получатель. Если в последней содержатся постоянные, отсутствующие в базе-источнике, их корреляциям с копируемыми константами приписывается значение "Null". Функция имеет следующий формат:

$$\text{Copy}[\text{const}, \text{db_source}, \text{db_receiver}],$$

где *const* может быть идентификатором копируемой постоянной или списком копируемых постоянных, а два последних параметра — имена базы-источника и базы-получателя соответственно.

Скопируем из основной базы в базу данных **wdb** постоянную Ридберга:

```
In[11] := wdb = Copy[ "rydbergConstant", dataBase, wdb ];
```

```
In[12] := PrintDB[ wdb[[1]] ]
```

Constant Table

Code	Name	Symbol	Value	Units
1	fineStructureConstant	α	$7.297\ 352\ 568(24) \times 10^{-3}$	
2	planckConstant	h	$6.626\ 0693(11) \times 10^{-34}$	J s
3	rydbergConstant	R_∞	10 973 731.568 525(73)	m^{-1}
4	speedOfLightInVacuum	c_0	299 792 458	m s^{-1}

```
In[13] := PrintDB[ wdb[[3]] ]
```

Correlation Coefficient Table

i	j	x_i	x_j	r_{ij}
1	2	α	h	0.010
1	3	α	R_∞	-0.017
1	4	α	c_0	0.000
2	3	h	R_∞	0.000
2	4	h	c_0	0.000
3	4	R_∞	c_0	0.000

Если копируемая постоянная уже существует в базе-получателе, копирование не выполняется, а функция выдает сообщение

Copy:: *constant_name* exists in DB-recipient already.

Так, в случае попытки повторного копирования постоянной Ридберга в базу данных **wdb** функция выдаст сообщение

Copy:: *rydbergConstant* exists in DB-recipient already.

4.1.7. Shift

Существующий по умолчанию алфавитный порядок расположения постоянных в базе данных может быть изменен с помощью функции **Shift**, которая имеет следующий формат:

`Shift[const, new_position, data_base],`

где *const* — идентификатор перемещаемой постоянной; *new_position* — новый код (позиция) константы; *data_base* — имя базы данных, где локализована эта постоянная. При выполнении функции **Shift** трансформируются все три элемента (таблицы) базы данных.

4.2. Функции доступа к данным

Функции этой группы извлекают необходимую при вычислениях информацию о физических постоянных, включенных в базу данных (средние значения, стандартные отклонения и коэффициенты корреляции).

Access	извлекает и представляет в числовом формате среднее значение, стандартное отклонение и относительную ошибку указанной постоянной.
PairCC	извлекает из таблицы коэффициентов корреляции строку, ассоциированную с указанной парой постоянных.
GroupCC	извлекает из таблицы коэффициентов корреляции субтаблицу корреляций одной постоянной с указанной группой постоянных.
MakeCM	преобразует таблицу коэффициентов корреляции указанной базы данных с исключенными точными константами в числовую матрицу корреляций.

4.2.1. Access

Эта функция извлекает и представляет в числовом формате среднее значение и стандартное отклонение постоянной из указанной базы данных, а также вычисляет и представляет относительную ошибку ее значения. Она имеет формат

`Access[data_base[[1]], "const_name"],`

где *data_base*[[1]] — первый элемент базы данных (таблица постоянных); *const_name* — пакетное имя постоянной в *character string* формате. Результат представляется списком

`{ mean_value, uncertainty, relative_uncertainty }.`

Смысл элементов этого списка зависит от *типа значения* постоянной. В текущей версии базы данных постоянные имеют четыре типа значений, которые условно назовем так:

- *точные*, например, значение магнитной постоянной ($\mu_0 = 4\pi \times 10^{-7} \text{ NA}^{-2}$);
- *статистически оцененные*, например значение постоянной Планка ($h = 6.6260693(11) \times 10^{-34} \text{ Js}$, то есть $[6.6260693 \pm 0.0000011] \times 10^{-34} \text{ Js}$);
- *интервально оцененные*, например плотность локального гало (*local halo density* $\rho_{\text{halo}} = (2-13) \times 10^{-25} \text{ g cm}^{-3}$);
- *неоцененные* (“псевдоточные”), например радиус Земли ($R_{\oplus} = 6.378140 \times 10^6 \text{ m}$).

Ниже приводятся примеры применения функции `Access` к постоянным всех четырех типов.

```
In[14] := Access[ basicTable, "magneticConstant"]
```

```
Out[14] = {  $\frac{\pi \text{ Second Volt}}{2500000 \text{ Ampere Meter}}$ , Exact, Exact }
```

```
In[15] := Access[ mydb[[1]], "planckConstant"]
```

```
Out[15] = {  $6.6260693 \times 10^{-34} \text{ Joule Second}$ ,  $1.1 \times 10^{-40} \text{ Joule Second}$ ,  $1.7 \times 10^{-7}$  }
```

Значения интервально оцененных постоянных функция `Access` трансформирует к виду статистически оцененных согласно правилам:

$$V_{\text{mean}} = \frac{V_{\text{max}} + V_{\text{min}}}{2}; \quad \sigma_V = \frac{V_{\text{max}} - V_{\text{min}}}{2}.$$

```
In[16] := Access[ basicTable, "localHaloDensity"]
```

```
Out[16] = {  $\frac{7.5 \times 10^{-25} \text{ Gram}}{\text{cMeter}^3}$ ,  $\frac{5.5 \times 10^{-25} \text{ Gram}}{\text{cMeter}^3}$ , 0.73 }
```

Значению неоцененных постоянных приписывается равная нулю неопределенность.

```
In[17] := Access[ basicTable, "earthRadius"]
```

```
Out[17] = {  $6.378140 \times 10^6 \text{ Meter}$ , 0, 0 }
```

4.2.2. PairCC

Эта функция извлекает из таблицы коэффициентов корреляций (третий элемент базы данных) *строку*, соответствующую указанной паре постоянных. Она имеет следующий формат:

`PairCC[const1, const2, data_base].`

Здесь первые два параметра — идентификаторы двух постоянных; *data_base* — имя базы данных, где хранится информация для этих констант. `In[18] := PairCC["bohrMagneton", "electronMass", dataBase]`

`Out[18] = { 8, 30, μ_B , m_e , 0.991 }.`

Здесь два первых элемента являются кодами постоянных, два следующих — их символьные обозначения, а последний элемент — коэффициент корреляции в *character string* формате. Для его численного представления можно воспользоваться стандартной процедурой:

`In[19] := ToExpression[Last[%]]`

`Out[19] = 0.991`

`In[20] := NumberQ[%]`

`Out[20] = True`

4.2.3. GroupCC

С помощью этой функции можно получить коэффициенты корреляции одной постоянной с группой констант из указанной базы данных. Функция имеет формат:

`GroupCC[const, {c1, c2, ...}, data_base],`
`GroupCC[const, {c1, c2, ...}, data_base, "print"].`

Здесь первый параметр — идентификатор физической постоянной, второй параметр представляет собой список идентификаторов группы констант; *data_base* — имя базы данных, где хранится информация по всем указанным постоянным.

`In[21] := GroupCC[8, {"electronMass", 38, "boltzmannConstant", 89}, dataBase]`

`Out[21] = {{8, 30, μ_B , m_e , 0.991}, {8, 38, μ_B , α , 0.106},`
`{8, 11, μ_B , k, 0.096}, {8, 89, μ_B , R_∞ , -0.002}}`

Третий необязательный параметр ("*print*") позволяет получить таблицу коэффициентов корреляции в табулированном виде.

`In[22] := GroupCC[8, {"electronMass", 38, "boltzmannConstant", 89}, dataBase, "print"];`

i	j	x_i	x_j	r_{ij}
8	30	μ_B	m_e	0.991
8	38	μ_B	α	0.106
8	11	μ_B	k	0.096
8	89	μ_B	R_∞	-0.002

4.2.4. MakeCM

Эта функция преобразует таблицу коэффициентов корреляции указанной базы данных с исключенными точными постоянными в числовую матрицу корреляций. Допустимые форматы функции:

```
MakeCM[ data_base ]
MakeCM[ data_base, "print" ],
```

где *data_base* — имя базы данных, а второй необязательный параметр (“*print*”) позволяет представить корреляционную матрицу в табулированной форме.

```
In[23] := MakeCM[ wdb ]
```

```
Out[23] = {{1., 0.01, -0.017}, {0.01, 1., 0.}, {-0.017, 0., 1.}}
```

Эта же матрица в табулированном виде выглядит так:

```
In[24] := MakeCM[ wdb, "print" ];
```

	α	h	R_∞
α	1.000	0.010	-0.017
h	0.010	1.000	0.000
R_∞	-0.017	0.000	1.000

Кроме того, функция позволяет строить корреляционную матрицу для указанного списка постоянных из **основной базы данных**, которая должна быть загружена. Формат функции при этом не изменяется, но в качестве первого параметра вводится список постоянных, а не имя базы данных.

```
In[25] := MakeCM[ {"muonMagneticMomentAnomaly",
"electronMagneticMoment", "muonMass"}, "print"];
```

	μ_e	a_μ	m_μ
μ_e	1.000	-0.001	-0.985
a_μ	-0.001	1.000	0.008
m_μ	-0.985	0.008	1.000

4.3. Функции—“калькуляторы”

Функции этой группы предназначены для вычисления средних значений алгебраических функций, зависящих от физических постоянных, оценки их вариаций и ковариаций с учетом неопределенностей и корреляций констант в соответствии со стандартной процедурой переноса ошибок.

ErrorPropagator	вычисляет среднее значение, стандартное отклонение и относительную ошибку функции от физических постоянных.
MatrixCharacteristics	вычисляет собственные значения, число обусловленности, детерминант и критерий округления элементов указанной матрицы, проверяет ее положительную определенность.
Covariator	вычисляет матрицы ковариаций и корреляций для вектора алгебраических функций, зависящих от физических постоянных.

4.3.1. ErrorPropagator

Эта функция предназначена для вычисления среднего значения, стандартного отклонения и относительной ошибки физической величины, которая представлена алгебраическим выражением, содержащим физические константы, с учетом неопределенностей значений последних и их корреляций. При этом функция осуществляет контроль качества входной матрицы корреляций, информируя пользователя о наличии отрицательных собственных значений, если таковые имеются.

Формат функции

```
ErrorPropagator[ expression, data_base ],
ErrorPropagator[ expression, data_base, {contr_keys} ],
```

где *expression* — выражение или переменная, которой оно присвоено; *data_base* — имя базы данных, содержащей информацию о всех случайных величинах, входящих в это выражение; *{contr_keys}* — список управляющих ключей, действие которых описывается ниже.

Среднее значение функции $\langle Q \rangle$, представленной выражением (*expression*), вычисляется стандартными средствами системы *Mathematica* с использованием средних значений постоянных, извлекаемых из базы данных. Поэтому все физические константы, входящие в выражение, должны быть представлены их пакетными именами в формате *symbol*. Вычисление ее стандартного отклонения $u(Q)$ основано на вычислении вариации $u(Q)^2$ в соответствии с известной статистической процедурой (см., например, [1], [9]):

$$u(Q)^2 = \sum_{i=1}^N \left(\frac{\partial Q}{\partial x_i} \right)^2 u(c_i)^2 + \sum_{i \neq j}^N \frac{\partial Q}{\partial x_i} \frac{\partial Q}{\partial x_j} u(c_i) u(c_j) r_{ij},$$

где частные производные вычисляются аналитически при значениях, $x_i = c_i$, $u(c_i)$ есть стандартное отклонение значения постоянной c_i , а r_{ij} — соответствующий элемент корреляционной матрицы \mathbf{r} . Если входная корреляционная матрица положительно полуопределена, функция **ErrorPropagator** выводит “стандартное” представление вычисленного значения физической величины:

{ mean_value, uncertainty, relative_uncertainty },

где стандартное отклонение (*uncertainty*) вычисляется как $u(Q) = \sqrt{u(Q)^2}$, а относительная неопределенность (*relative_uncertainty*) — как $u_r(Q) = u(Q)/\langle Q \rangle$, $\langle Q \rangle \neq 0$.

Если входная корреляционная матрица (\mathbf{r}) имеет отрицательные собственные значения, функция выполняет вычисление неопределенности искомой величины как с входной, так и с единичной матрицей корреляций. В этом случае выводимый список содержит входную матрицу корреляций и два значения искомой величины, полученных с входной и единичной матрицей соответственно:

$$\begin{aligned} & \{ \{non\text{-}positive\ semidefinite\ submatrix\}, \\ & \{ \{mean_value, uncertainty, relative_uncertainty\}, \\ & \{mean_value, uncert_1, relative_uncert_1\} \} \}. \end{aligned}$$

Третий (необязательный) параметр функции `ErrorPropagator` в общем случае является списком двух элементов – “ключей”. Первый ключ управляет изменением элементов входной матрицы корреляций. Второй элемент (ключ) регулирует выходную информацию. Их допустимые значения приведены в следующей таблице:

<code>contr_keys[[1]]</code>	Действие
Отсутствует	всем элементам входной матрицы корреляций со значением “Null” присваивается нулевое значение (0).
0	входная корреляционная матрица заменяется на единичную матрицу с тем же числом компонент.
“Null”	все неизвестные элементы матрицы корреляции $r_{ij} = \text{“Null”}$ заменяются пакетной функцией <code>cS[i,j]</code> , которой затем можно присваивать любые значения.
<code>{ {i, j, value}, ... }</code>	корректирует значения указанных элементов входной матрицы корреляции (r_{ij}), присваивая им новые значение (<code>value</code>).
<code>contr_keys[[2]]</code>	Действие
Отсутствует	стандартный вывод: <code>{ mean_value, uncertainty, relative_uncertainty }</code> .
“CM”	стандартный вывод дополняется входной матрицей корреляций.
“EM”	стандартный вывод дополняется “матрицей ошибок”, элементами которой являются отдельные слагаемые в выражении для $\mathbf{u}(\mathbf{Q})^2$.
“All”	стандартный вывод дополняется обеими матрицами.

В качестве примера применения функции `ErrorPropagator` приведем вычисление магнетона Бора по двум формулам:

$$\mu_B = \left(\frac{c_0 \alpha^5 h}{32 \pi^2 \mu_0 R_\infty^2} \right)^{1/2} = \frac{e h}{4 \pi m_e}.$$

Это позволит продемонстрировать ее работу с различными входными матрицами корреляций.

```
In[26] := bm1 = Sqrt[speedOfLightInVacuum fineStructureConstant^5 *
                planckConstant / (32 \pi^2 magneticConstant rydbergConstant^2)];
```

```
In[27] := bv1 = PowerExpand[ ErrorPropagator[ bm1, dataBase ] /. Volt ->
                Joule Ampere^-1 Second^-1 /. Ampere -> Joule Tesla^-1 Meter^-2 ]
```

```
Out[27] = {  $\frac{9.2740095 \times 10^{-24} \text{ Joule}}{\text{Tesla}}$ ,  $\frac{7.74319 \times 10^{-31} \text{ Joule}}{\text{Tesla}}$ ,  $8.34935 \times 10^{-8}$  }
```

Здесь и далее численные результаты представляются в стандартном формате системы *Mathematica*. Используя функцию `SetPrecision`, можно получить больше цифр:

```
In[28] := SetPrecision[First[bv1] * Tesla/Joule, 9 ] * Joule/Tesla
```

```
Out[28] =  $\frac{9.27400948 \times 10^{-24} \text{ Joule}}{\text{Tesla}}$ 
```

Вычисление по второй формуле дает следующий результат:

```
In[29] := bm2 = elementaryCharge planckConstant / (4 π electronMass);
```

```
In[30] := bv2 = ErrorPropagator[ bm2, dataBase ] /. Coulomb ->
                                             kGram Tesla-1 Second-1
```

```
WARNING: correlation matrix is non-positive semidefinite.
```

```
Minimum eigenvalue of this matrix is -0.000154812
```

```
Out[30] = { {{1., 0.998, 0.999}, {0.998, 1., 1.}, {0.999, 1., 1.}},
            { {  $\frac{9.274009 \times 10^{-24} \text{ Joule}}{\text{Tesla}}$ ,  $\frac{7.28149 \times 10^{-31} \text{ Joule}}{\text{Tesla}}$ ,  $7.85151 \times 10^{-8}$  },
              {  $\frac{9.274009 \times 10^{-24} \text{ Joule}}{\text{Tesla}}$ ,  $\frac{2.38336 \times 10^{-30} \text{ Joule}}{\text{Tesla}}$ ,  $2.56993 \times 10^{-7}$  } } }
```

```
In[31] := SetPrecision[ First[First[Last[bv2]]] * Tesla/Joule, 9 ] * Joule/Tesla
```

```
Out[31] =  $\frac{9.27400941 \times 10^{-24} \text{ Joule}}{\text{Tesla}}$ 
```

Как видно из результатов вычислений, нам не удастся воспроизвести значение магнетона Бора, представленного на сайте NIST [2]:

$$\mu_B = 9.274\,009\,49(80) \times 10^{-24} \text{ J T}^{-1}.$$

Отметим, что эта неудача не связана с точностью численных процедур в *Mathematica*. Причиной неудачи является недостаточная точность представления коэффициентов корреляции в [2]. Более отчетливо это демонстрируют результаты вычислений по второй формуле. В этом случае функция предупредила о неположительной полуопределенности входной корреляционной матрицы, взятой из [2]. При этом, как уже было сказано выше, представленный результат включает эту матрицу и два значения искомой величины; первое значение получено с использованием некорректной матрицы корреляций, и мы его далее не обсуждаем, а второе — с единичной матрицей. Как видно, в случае вычислений с единичной матрицей мы получаем втрое завышенное значение стандартного отклонения по сравнению с рекомендуемым значением.

С помощью управляющих ключей и в первом случае можно, например, выполнить вычисление с единичной матрицей или вывести матрицу корреляций с тем, чтобы в дальнейшем изучить ее свойства:

```
In[32] := bv1 = PowerExpand[ ErrorPropagator[ bm1, dataBase, {"CM"} ] /.
                               Volt -> Joule Ampere-1 Second-1 /. Ampere -> Joule Tesla-1 Meter-2 ]
```

$$\text{Out}[32] = \{\{\{1., 0.01, -0.017\}, \{0.01, 1., 0.\}, \{-0.017, 0., 1.\}\}, \\ \left\{ \frac{9.2740095 \times 10^{-24} \text{ Joule}}{\text{Tesla}}, \frac{7.74319 \times 10^{-31} \text{ Joule}}{\text{Tesla}}, 8.34935 \times 10^{-8} \right\}$$

Примеры исследования некоторых характеристик корреляционных матриц и, в частности, матриц из рассмотренных примеров приводятся в следующем разделе.

4.3.2. MatrixCharacteristics

Эта функция позволяет одновременно определять такие характеристики числовой квадратной матрицы, как ее собственные значения, число обусловленности и детерминант. Она используется описываемой ниже функцией `Covariator` для контроля качества матриц корреляций, но может оказаться полезной и для внешнего применения. Функция имеет следующий формат:

```
MatrixCharacteristics[ matrix ],
MatrixCharacteristics[ matrix, "matrix_name" ] .
```

Выходная информация функции представляется списком

$$\{ \text{condition_number}, \text{determinant}, \text{rounding_index}, \text{eigenvalues_list} \}.$$

Используя штатную функцию `Eigenvalues[matrix]` системы *Mathematica*, функция `MatrixCharacteristics` находит собственные числа указанной матрицы (λ_i , $i = \overline{1, n}$; где n — размерность $n \times n$ -матрицы), а затем вычисляет ее число обусловленности $|\lambda_{\max}|/|\lambda_{\min}|$ и детерминант как $\prod_{i=1}^n \lambda_i$. Кроме того, для положительно определенных матриц вычисляется индекс независимого округления (I_{\min}) ее элементов [11]:

$$I_{\min} = \left\lceil \log_{10} \left(\frac{n-1}{2 \lambda_{\min}} \right) \right\rceil.$$

I_{\min} указывает минимальное число цифр после десятичной точки, которое необходимо сохранять при независимом округлении элементов матрицы, чтобы она не потеряла положительную определенность. В случае, если матрица оказывается неположительно определенной, индекс не вычисляется, а функция выводит предупреждение.

Рассмотрим примеры работы функции с входными корреляционными матрицами из двух предыдущих примеров. При вычислении магнетона Бора по первой формуле входная матрица имеет вид

	α	h	R_{∞}
α	1.000	0.010	-0.017
h	0.010	1.000	0.000
R_{∞}	-0.017	0.000	1.000

и следующие характеристики:

```
In[33] := MatrixCharacteristics[ First[bv1], "correlation matrix (1-st formula)"]
```

Characteristics of correlation matrix (1-st formula)

- 1) Condition number = 1.04024
- 2) Determinant = 0.999611
- 3) Rounding criterion: $I_{\min} = 1$

`Out[33] = {1.04024, 0.999611, 1, {1.01972, 1., 0.980277}}`

Входная матрица во втором случае (вторая формула) имеет вид

	m_e	e	h
m_e	1.000	0.998	0.999
e	0.998	1.000	1.000
h	0.999	1.000	1.000

и по информации функции `ErrorPropagator` не является положительно определенной.

```
In[34] := MatrixCharacteristics[ First[bv2], "correlation matrix (2-d formula)"];
```

Characteristics of correlation matrix (2-d formula)

- 1) Condition number = 19365.5
- 2) Determinant = $-1. \times 10^{-6}$
- 3) WARNING: the matrix is non-positive semidefinite
(Minimum eigenvalue of this matrix is -0.000154812)

В этом примере, закончив ввод функции символом “;”, мы ограничились получением лишь текстовой информации, выводимой автоматически.

4.3.3. Covariator

Эта функция дополняет функцию `ErrorPropagator`, позволяя вычислять ковариационную и корреляционную матрицы для неопределенностей компонент вектор-функций, зависящих от физических постоянных. Функция `Covariator` имеет следующий формат:

```
Covariator[ expr_list, data_base ],  
Covariator[ expr_list, data_base, {contr_keys} ],
```

где `expr_list` — список выражений для вычисляемых (выходных) величин (как правило, представленный переменной, которой этот список присвоен), имеющий следующую структуру:

$$exprlist = \{ \{outputc_1, expr_1\}, \{outputc_2, expr_2\}, \dots, \{outputc_n, expr_n\} \}.$$

По сути этот список есть набор соотношений: $outputc_1 = expr_1$, $outputc_2 = expr_2$, ..., среди которых могут быть и тождества ($constant_i \equiv constant_i$).

Второй параметр функции (`data_base`) — имя базы данных, содержащей информацию как о входных величинах (включая рекомендованные ФФП и их корреляции), входящих в выражения $expr_i$, так и о вычисляемых (выходных) величинах ($outputc_i$). Причем для последних допускается отсутствие информации *только* для коэффициентов корреляции, а их средние значения и неопределенности должны быть включены в эту базу данных. Если эти величины не известны, их можно получить с помощью функции `ErrorPropagator`, а затем включить в базу данных ².

²Неудобство, связанное с необходимостью включения информации о выходных величинах в рабочую базу данных, предполагается устранить уже в следующей версии пакета.

Последний (необязательный) параметр `{contr_keys}` содержит управляющие ключи, действие которых описывается ниже.

Функция `Covariator` работает в соответствии со стандартной статистической процедурой “переноса неопределенностей” (см., например, [1, 9]) входных величин \mathbf{c}_i ($i = 1, 2, \dots, M$) (независимые переменные) на вычисляемые физические величины (зависимые переменные) $P_j(\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_M)$, где $j = 1, 2, \dots, N$; $N \geq M$:

$$u(P_k, P_l) = \sum_{i,j=1}^M \frac{\partial P_k}{\partial x_i} \frac{\partial P_l}{\partial x_j} u(\mathbf{c}_i) u(\mathbf{c}_j) r_{ij}.$$

Здесь частные производные вычисляются аналитически при значениях $\mathbf{x}_i = \mathbf{c}_i$, $u(\mathbf{c}_i)$ — стандартное отклонение значения согласованной постоянной \mathbf{c}_i , а r_{ij} — соответствующий элемент корреляционной матрицы. В действительности функция вычисляет *относительные* ковариации ($\mathbf{u}_r(P_k, P_l) = u(P_k, P_l) / (P_k P_l)$) и вариации ($\mathbf{u}_r^2(P_k) = u_r(P_k, P_k) = u(P_k, P_k) / P_k^2$), а затем коэффициенты корреляции:

$$r(P_k, P_l) = \frac{u_r(P_k, P_l)}{\sqrt{u_r^2(P_k) u_r^2(P_l)}}.$$

Функция `Covariator` тестирует входную матрицу корреляций на положительную полуопределенность, используя для этого функцию `MatrixCharacteristics`. Если матрица положительно полуопределена, функция выполняет стандартную процедуру вычисления корреляционной матрицы неопределенностей вычисляемых физических величин из указанной базы данных. В противном случае, как и в `ErrorPropagator`, вычисление ковариаций и корреляций осуществляется дважды — с входной и единичной матрицами. Таким образом, выводимый результат включает две выходных матрицы корреляций. Кроме того, функция автоматически выводит некоторую дополнительную информацию: списки входных и выходных величин, а также их матрицы корреляции в табулированном виде.

В качестве примера рассмотрим вычисление коэффициентов корреляции стандартного отклонения магнетона Бора, с отклонениями ФФП, входящих в первую формулу (см. раздел 4.3.1), которая в терминах пакетных имен постоянных имеет вид

```
In[35] := bm = Sqrt[speedOfLightInVacuum fineStructureConstant5 *
                planckConstant / (32 π2 magneticConstant rydbergConstant2)];
```

Теперь осуществим ввод списка (вектора) функций от констант:

```
In[36] := listbm = { {bohrMagneton, bm},
                    {fineStructureConstant, fineStructureConstant},
                    {planckConstant, planckConstant},
                    {rydbergConstant, rydbergConstant} };
```

И, наконец, иницилируем операцию вычисления ковариаций (корреляций)

```
In[37] := cmbm = Covariator[ listbm, dataBase ]
```

Input constants: {fineStructureConstant, magneticConstant,
planckConstant, rydbergConstant, speedOfLightInVacuum}
Output constants: {bohrMagneton, fineStructureConstant, planckConstant,
rydbergConstant}

	α	h	R_∞
α	1.000	0.010	-0.017
h	0.010	1.000	0.000
R_∞	-0.017	0.000	1.000

Characteristics of input correlation matrix

- 1) Condition number = 1.04024
- 2) Determinant = 0.999611
- 3) Rounding criterion: $I_{min} = 1$

	μ_B	α	h	R_∞
μ_B	1.	0.10842	0.99514	-0.00175378
α	0.10842	1.	0.01	-0.017
h	0.99514	0.01	1.	0.
R_∞	-0.00175378	-0.017	0.	1.

Characteristics of output correlation matrix

- 1) Condition number = 6.01129×10^{15}
- 2) Determinant = 6.65223×10^{-16}
- 3) Rounding criterion: $I_{min} = 16$

Out[37] = { $\{\mu_B, \alpha, h, R_\infty\}$,
 $\{1., 0.10842, 0.99514, -0.00175378\}$, $\{0.10842, 1., 0.01, -0.017\}$,
 $\{0.99514, 0.01, 1., 0.\}$, $\{-0.00175378, -0.017, 0., 1.\}$ }

Прокомментируем выходную информацию в порядке ее следования. Список входных констант оказался больше списка выходных величин, так как в него включены две точные константы. После корреляционной матрицы входных постоянных и ее характеристик представлена выходная матрица с вычисленными коэффициентами корреляций для магнетона Бора, которую сравним с рекомендованной матрицей:

	μ_B	α	h	R_∞
μ_B	1.	0.106	0.995	-0.002
α	0.106	1.	0.01	-0.017
h	0.995	0.01	1.	0.
R_∞	-0.002	-0.017	0.	1.

Characteristics of the recommended correlation matrix

- 1) Condition number = 5343.7
- 2) Determinant = 0.000748049
- 3) Rounding criterion: $I_{min} = 4$

Соответствующие матричные элементы обеих матриц мало отличаются друг от друга (на первый взгляд), однако характеристики вычисленной нами матрицы оказались гораздо хуже, чем у рекомендованной матрицы. Мы связываем эту ситуацию с разной последовательностью операций вычисления и округления в этих двух случаях. Следует отметить, что приведенная выше рекомендованная подматрица корреляций оказалась “переокругленной”, так как для нее $I_{min} = 4$.

Третий (необязательный) параметр функции `Covariator` в общем случае является списком двух элементов. Первый ключ управляет изменением элементов входной корреляционной матрицы точно так же, как и в функции `ErrorPropagator`. Второй элемент (ключ) регулирует выходную информацию и может принимать следующие значения:

<code>contr_keys[[2]]</code>	Действие
Отсутствует	стандартный вывод корреляционной матрицы.
“CVM”	вместо корреляционной матрицы выводится матрица относительных ковариаций.
“Mix”	выводится комбинированная матрица, где на главной диагонали представлены относительные вариации, выше нее — относительные ковариации, а ниже — коэффициенты корреляции.
“All”	выводятся все три матрицы.

Заметим, что элементы матрицы относительных ковариаций масштабируются по ее минимальному элементу. Масштабирующий множитель выводится сразу после матрицы. Таким образом, умножение полученной матрицы на этот фактор дает истинную матрицу относительных ковариаций.

5. Заключение

Как уже говорилось во введении, пакет `StandardPhysicalConstants` находится в состоянии “становления”. Ближайшие планы его развития предусматривают решение следующих проблем:

- Проблемы обеспечения качества данных в нашей базе ФФП, что, в первую очередь, подразумевает их представление в виде корректно округленных десятичных чисел. К сожалению, авторы не получили ответа от издателей рекомендованных данных на предложения по исправлению некорректных данных и оптимизации веб-доступа к ним. По-видимому, решение этой проблемы может быть получено на пути независимой оценки и согласования ФФП [11].
- Проблемы организации корректной и комфортной работы с размерными величинами на внутреннем (пакетном) и внешнем (пользовательском) уровнях.
- Проблемы контроля корректности вычислений на сложных структурах данных, используемых в пакете.

Список литературы

- [1] Mohr, P. and Taylor B. *CODATA recommended values of the fundamental physical constants: 1998*, in Rev. Mod. Phys., **72**, 351, (April 2000), The American Physical Society; <http://physics.nist.gov/cuu/Constants/archive1998.html>
К сожалению, в архивной версии (V.3.2) не сохранены коэффициенты корреляций.
- [2] Mohr, P. and Taylor B. *CODATA recommended values of the fundamental physical constants: 2002*. <http://physics.nist.gov/cuu/Constants/>.
- [3] Ezhela, V., Larin, V. *The Development of the Mathematica Package 'StandardPhysicalConstants'*, in Proc. of the 5th International *Mathematica* Symposium (eds. P.Mitic, Ph.Ramsden, J.Carne), Imperial College Press, London, 2003, pp. 207–214; IHEP Preprint 2003-17, Protvino, 2003.
- [4] <http://library.wolfram.com/howtos/constants/>.
- [5] <http://www.mapleapps.com/categories/maple8/html/scientificconstants.html>.
- [6] The MAXIMA computer algebra system, “physconst” package. <http://maxima.sourceforge.net/>
- [7] Ezhela, V. and Larin, V. *The physical constants for Mathematica*, in Proc. of the fourth International *Mathematica* Symposium (eds. Y.Tazawa, S.Sakakibara, S.Ohashi, Y.Uemura) Tokyo Denki University Press, June 2001, pp. 263–268; IHEP Preprint 2001-27, Protvino, 2001.
- [8] Wolfram, S. *The Mathematica Book*. 4th ed. Wolfram Media/Cambridge University Press, 1999.
- [9] S. Eidelman, K.G. Hayes, K.A. Olive et al., *Review of particle physics* [Particle Data Group], Phys. Lett. **B592** (2004) 1.
- [10] Hagiwara, K. et al., *Review of particle physics*. [Particle Data Group], in Phys. Rev., **D66**, 010001-1, (July 2002), The American Physical Society.
- [11] Siver, A.S. and Ezhela, V.V. *On the CODATA Recommended Values of the Fundamental Physical Constants: V.3.2(1998) & V.4.0(2002)*. IHEP Preprint 2003-34, Protvino, 2003.

Рукопись поступила 29 июня 2004 г.

6. Приложение

Имена постоянных в Dbfrc113.m

Код	Пакетное имя (<i>basicTable</i>)	Стандартное имя (<i>standardNames</i>)
1	accelerationOfGravity	standard acceleration of gravity
2	ageOfUniverse	age of the universe
3	ageOfUniverseInGYear	age of the universe in Gyear
4	astronomicalUnit	astronomical unit (mean \oplus – \odot distance)
5	atomicMassConstant	atomic mass constant
6	avogadroConstant	Avogadro constant
7	baryonDensityOfUniverse	baryon density of the universe
8	bohrMagneton	Bohr magneton
9	bohrMagnetonIneV	Bohr magneton in eV/T
10	bohrRadius	Bohr radius
11	boltzmannConstant	Boltzmann constant
12	classicalElectronRadius	classical electron radius
13	comptonWavelength	Compton wavelength
14	comptonWavelengthReduced	Compton wavelength over 2π
15	conductanceQuantum	conductance quantum
16	constantOfGravitation	Newtonian constant of gravitation
17	conversionConstant	Planck constant over 2π times c in $MeV\ fm$
18	conversionConstantSquare	conversion constant square
19	cosmicBackgroundTemperature	present day CBR temperature
20	deuteronMagneticMoment	deuteron magnetic moment
21	deuteronMass	deuteron mass
22	earthMass	Earth mass
23	earthRadius	Earth mean equatorial radius
24	earthSchwarzschildRadius	Schwarzschild radius of the Earth
25	electricConstant	electric constant
26	electronChargeToMassQuotient	electron charge to mass quotient
27	electronGFactor	electron g -factor
28	electronMagneticMoment	electron magnetic moment
29	electronMagneticMomentAnomaly	electron magnetic moment anomaly
30	electronMass	electron mass
31	electronMassInMeV	electron mass energy equivalent in MeV
32	electronMuonMagneticMomentRatio	electron-muon magnetic moment ratio
33	electronMuonMassRatio	electron-muon mass ratio
34	elementaryCharge	elementary charge
35	eVToKilogramRelationship	electron volt-kilogram relationship
36	faradayConstant	Faraday constant
37	fermiCouplingConstant	Fermi coupling constant
38	fineStructureConstant	fine-structure constant

Продолжение

Код	Пакетное имя (<i>basicTable</i>)	Стандартное имя (<i>standardNames</i>)
39	galacticUnit	solar distance from galactic center
40	galacticUnitInParsec	solar distance from galactic center in parsec
41	galaxyVelocityWithRespectToCBR	Local group velocity with respect to CBR
42	hubbleConstant	present day Hubble expansion rate in SI
43	icePoint	ice point (standard temperature)
44	impedanceOfVacuum	characteristic impedance of vacuum
45	inverseFineStructureConstant	inverse fine-structure constant
46	jansky	Jansky
47	localDiskDensity	local disk density
48	localDiskDensityInGeV	local disk density in <i>GeV</i>
49	localHaloDensity	local halo density
50	localHaloDensityInGeV	local halo density in <i>GeV</i>
51	magneticConstant	magnetic constant
52	magneticFluxQuantum	magnetic flux quantum
53	meanSiderealDay	mean sidereal day (2005.0)
54	molarGasConstant	molar gas constant
55	molarVolume	molar volume of ideal gas (273.15K, 101.325 kPa)
56	muonComptonWavelength	muon Compton wavelength
57	muonComptonWavelengthReduced	muon Compton wavelength over 2π
58	muonElectronMassRatio	muon-electron mass ratio
59	muonGFactor	muon <i>g</i> -factor
60	muonMagneticMoment	muon magnetic moment
61	muonMagneticMomentAnomaly	muon magnetic moment anomaly
62	muonMagneticMomentToBohrMagnetron	muon magnetic moment to Bohr magneton ratio
63	muonMass	muon mass
64	muonMassInMeV	muon mass energy equivalent in <i>MeV</i>
65	muonProtonMagneticMomentRatio	muon-proton magnetic moment ratio
66	muonProtonMassRatio	muon-proton mass ratio
67	neutronComptonWavelength	neutron Compton wavelength
68	neutronMagneticMoment	neutron magnetic moment
69	neutronMass	neutron mass
70	normalizedBaryonDensity	baryon-to-photon ratio
71	nuclearMagnetron	nuclear magneton
72	nuclearMagnetronIneV	nuclear magneton in <i>eV/T</i>
73	parsec	parsec (1 AU/1 arc sec)
74	planckConstant	Planck constant
75	planckConstantInMeV	Planck constant in <i>eV s</i>
76	planckConstantReduced	Planck constant over 2π
77	planckConstantReducedInMeV	Planck constant over 2π in <i>eV s</i>

Продолжение

Код	Пакетное имя (<i>basicTable</i>)	Стандартное имя (<i>standardNames</i>)
78	planckLength	Planck length
79	planckMass	Planck mass
80	planckMassInGeV	Planck mass energy equivalent in <i>GeV</i>
81	planckTime	Planck time
82	pressurelessMatterDensity	pressureless matter density of the universe
83	protonChargeToMassQuotient	proton charge to mass quotient
84	protonComptonWavelength	proton Compton wavelength
85	protonMagneticMoment	proton magnetic moment
86	protonMagneticMomentToBohrMagneton	proton magnetic moment to Bohr magneton ratio
87	protonMass	proton mass
88	protonNeutronMassRatio	proton-neutron mass ratio
89	rydbergConstant	Rydberg constant
90	rydbergEnergy	Rydberg constant times hc in <i>eV</i>
91	rydbergFrequency	Rydberg constant times c in <i>Hz</i>
92	sackurTetrodeConstant	Sackur-Tetrode constant ($1K, 101.325 kPa$)
93	scaledCosmologicalConstant	dark energy density
94	siderealYear	sidereal year (fixed star to fixed star) (2005.0)
95	solarConstant	mean total solar irradiance (TSI)
96	solarLuminosity	solar luminosity
97	solarMass	solar mass
98	solarRadius	solar equatorial radius
99	solarSchwarzschildRadius	Schwarzschild radius of the Sun
100	solarVelocityAroundGalaxyCenter	solar velocity around center of Galaxy
101	solarVelocityWithRespectToCBR	solar velocity with respect to CBR
102	speedOfLightInVacuum	speed of light in vacuum
103	speedOfSound	speed of sound
104	stefanBoltzmannConstant	Stefan-Boltzmann constant
105	strongCouplingConstant	strong coupling constant
106	tauMass	tau mass
107	thomsonCrossSection	Thomson cross section
108	tropicalYear	tropical year (equinox to equinox) (2005.0)
109	wavelengthOfeVOverc0Particle	wavelength of $1eV/c$ particle
110	wBosonMass	W^\pm boson mass
111	weakMixingAngle	weak mixing angle
112	wienDisplacementLawConstant	Wien displacement law constant
113	zBosonMass	Z^0 boson mass

В.В. Ежела, В.Н. Ларин.
Пакет StandardPhysicalConstants 2.1. Руководство пользователя.

Оригинал-макет подготовлен с помощью системы **ИТ_EX**.
Редактор Н.В. Ежела. Технический редактор

Подписано к печати 30.06.2004. Формат 60 × 84/8. Офсетная печать.
Печ.л. 3,12. Уч.-изд.л. 2,5. Тираж 100. Заказ 281. Индекс 3649.

ГНЦ РФ Институт физики высоких энергий
142284, Протвино Московской обл.

