

УДК 5530.12+531.51+517.944+519.713+514.774

*Ю. Г. Игнатьев¹, В. А. Бушкова²***КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ И ОСНАЩЕННАЯ ДИНАМИЧЕСКАЯ ВИЗУАЛИЗАЦИЯ КРИВЫХ В ДИФФЕРЕНЦИАЛЬНОЙ, РИМАНОВОЙ ГЕОМЕТРИИ И ТЕОРИИ ГРАВИТАЦИИ**

Описаны методы компьютерного моделирования основных объектов дифференциальной геометрии и принципы оснащенной динамической визуализации основных объектов дифференциальной геометрии в прикладном математическом пакете Maple. Приведены примеры создания библиотек программных процедур оснащенных управляемых визуальных динамических моделей основных объектов дифференциальной и римановой геометрии.

Ключевые слова: математическое и компьютерное моделирование, оснащенная, управляемая динамическая визуализация, дифференциальная и риманова геометрия, теория гравитации, прикладные математические пакеты

PACS: 04.20.Cv, 98.80.Cq, 96.50.S 52.27.Ny

Введение

В прикладных геометрических задачах, в частности, в теории гравитации, основанной на римановой геометрии, все чаще приходится обращаться к методам численного моделирования, так как соответствующие задачи исследования являются либо нелинейными, либо чрезвычайно громоздкими. При этом поневоле приходится отказываться от аналитических методов исследования, обладающих высокой степенью прозрачности результатов, но при этом имеющих весьма узкий диапазон применения. Вследствие этого анализ результатов исследования и их адекватная интерпретация чрезвычайно затруднены, что зачастую не дает правильно понять существо исследуемого объекта или процесса. Все это вынуждает исследователя прибегать к методам математического и компьютерного моделирования исследуемого объекта [1] и визуализации математических моделей.

Системы компьютерной математики (СКМ) обладают богатыми и до сих пор еще мало оцененными графическими возможностями, позволяющими создавать многопараметрические графические модели как геометрических, так и физических объектов. Особенно богатыми возможностями трехмерной графики обладает пакет Maple³, предоставляющий пользователю возможность интерактивного взаимодействия с трехмерной графической средой, что открывает уникальные возможности изучения свойств трехмерных графических моделей. При этом следует отметить, что прямое использование стандартных графических процедур компьютерной математики не позволяет создавать достаточно сложные графические модели, - для создания их необходимо создавать собственные специализированные библиотеки программных графических процедур. Для создания этих процедур в нашей работе применяются расширенные опции процедур ядра Maple и библиотеки plots. Ниже формулируются основные принципы создания программных графических процедур в СКМ.

В статье описаны авторские вычислительные процедуры в пакете Maple решения стандартных задач дифференциальной геометрии. К стандартным задачам дифференциальной геометрии следует отнести задачи, полностью построенные на аппарате векторной алгебры и дифференциального исчисления функций одной (дифференциальная геометрия кривых) и двух (дифференциальная геометрия поверхностей) переменных. К таковым задачам относятся: построение адаптированного репера кривой, вычисление кривизны и кручения кривой, нахождение первой и второй квадратичных форм поверхностей, ее векторной оснастки, геодезической, нормальной, главных кривизн поверхностей. Именно эти задачи и изучаются в стандартных курсах дифференциальной геометрии. К задачам повышенной сложности, которые за редким исключением практически не изучаются в стандартных курсах дифференциальной геометрии относятся такие, как восстановление кривой по ее натуральным уравнениям и нахождение линий главных направлений кривизн

¹ E-mail: ignatev_yu@rambler.ru

² E-mail: vbushkova@inbox.ru

³ см., например, [2–5].

и геодезических на поверхностях. Замечательным является тот факт, что как раз на решения этих задач, как основных, и направлены стандартные курсы дифференциальной геометрии. Не решаются же эти заявленные и очень поучительные задачи в стандартных курсах дифференциальной геометрии именно потому, что задачи эти связаны с решением системы обыкновенных существенно нелинейных дифференциальных уравнений, для которых не существует общей теории интегрирования, за исключением некоторых качественных методов исследования. Эти задачи также могут быть решены средствами компьютерной математики. В конце 90-х годов А. Грэй посвятил специальную монографию изложению дифференциальной геометрии с помощью пакета компьютерной математики «Mathematica 2» [6]. Однако, в этой книге не отражены вопросы, как динамической визуализации объектов дифференциальной геометрии, так и вопросы, связанные с интегральными нелинейными задачами дифференциальной геометрии.

1. Принципы оснащенной динамической визуализации математических моделей в системе компьютерной математики «Maple»

1.1. Понятие об оснащенной динамической визуализации

Следует заметить, что пакет компьютерной математики Maple, начиная уже с самых ранних версий имеет несомненные преимущества в области 3D-графики, особенно, интерактивной и динамической, по сравнению с пакетом «Mathematica», и позволяет программными средствами решить указанные задачи⁴. Эта статья посвящена разработке программных процедур в пакете «Maple», позволяющих осуществлять управляемую, оснащенную динамическую визуализацию основных задач дифференциальной геометрии кривых.

Под управляемой, оснащенной динамической визуализацией здесь и далее мы понимаем визуализацию многопараметрической математической модели, изменение свойств которой можно наблюдать во временной последовательности в графической форме, сопровождаемой изменяющейся со временем числовой или графической информацией, с возможностью изменения пользователем параметров модели.

Методы оснащенной динамической визуализации математических моделей разрабатываются в группе Ю.Г. Игнатьева с 2004 года [8]. Отметим, что профессором М.Н. Кирсановым (Московский энергетический университет) разрабатываются методы динамической визуализации сложных, лагранжевых механических систем [9]. В работах [10–13] конструируются математические и компьютерные модели *нелинейных обобщенно-механических систем*, описываемых системой нелинейных обыкновенных дифференциальных уравнений произвольного порядка с помощью методов сплайновой интерполяции численных решений дифференциальных уравнений. Технология создания оснащенной управляемой динамической модели в СКМ Maple заключается в следующем [10, 14, 15] (см. Рис.). Сначала с помощью стандартных процедур Maple или программирования в пакете создаются графические объекты a_{ik} . Эти объекты могут быть как статическими графическими объектами, так и сложными, анимационными, созданными с помощью встроенных в Maple процедур анимации. Простые графические объекты представлены на Рис. прямоугольниками с горизонтально-градиентной заливкой, сложные - прямоугольниками с вертикальными секциями и вертикальной градиентной заливкой. Полученные графические объекты объединяются в i -тый кадр анимации, $frame_i$, с помощью встроенной процедуры `plots[display](a_i1, a_i2, ..., a_ini)`, с присвоением имени $A[i]$. Наконец, все кадры анимации объединяются в анимационную последовательность с помощью той же процедуры, но с добавочной опцией: `plots[display]([A[1], A[2], ..., A[n]], insequence=true)`:

В результате получается, вообще говоря, трехмерная, интерактивная анимационная картина, состоящая из n кадров. Дискретной временной переменной t при этом является номер кадра i , длительностью анимации, как и степенью ее непрерывности, управляет параметр n - число кадров. При создании анимации необходимо выбирать золотую середину между качеством анимации (большие значения n) и скоростью ее загрузки (малые значения n). Управление скоростью анимации, ее направлением, как и переходом в режим покадрового просмотра, можно осуществить непосредственно из графического меню окна Maple после выделения графического объекта. Для

⁴О сравнительных характеристиках систем «Mathematica» и «Maple» см., например, [5, 7].

управления же параметрами компьютерной модели эти параметры должны быть введены в программные процедуры графических объектов. Такими параметрами могут быть, например, уравнения линии, способ ее представления, тип ее оснащения, количество кадров анимации, предельные значения параметров кривой и т.п.

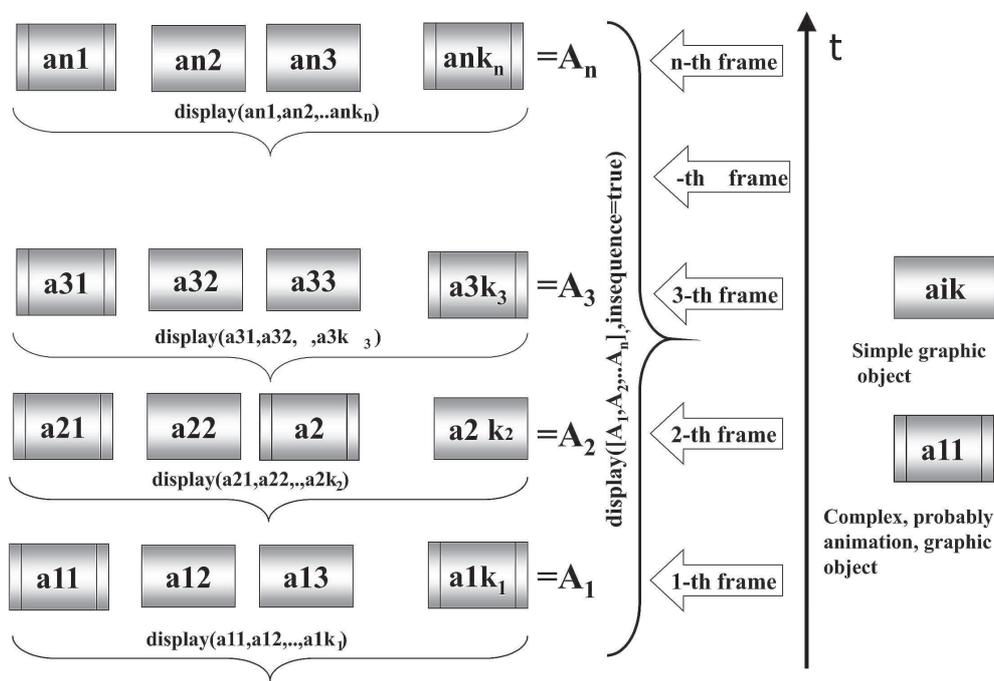


Рис. 1. Схема создания оснащенной динамической визуализации компьютерной модели.

Динамическое оснащение кривой, как и других графических объектов, может быть трех типов:

1. Графическое оснащение - оснащение с помощью дополнительных графических объектов: изображением точек, касательных и других векторов, плоскостей и т.п.;
2. Текстовое оснащение - оснащение с помощью динамических текстовых вставок;
3. Цифровое оснащение - оснащение с помощью динамических цифровых вставок.

При этом первые два типа динамического оснащения достигаются простым добавлением графических или текстовых объектов в каждый кадр с введением их функциональной зависимости от номера кадра, i , с помощью процедуры `display`. Цифровые динамические вставки величины s достигаются добавлением в каждый кадр процедуры вычисления этой величины, $S(i)$, которую предварительно необходимо конвертировать в строковую переменную: `convert(s=S(i), string)`. При этом данная величина будет отображаться в форме $s=SD(i)$, где $SD(i)$ - динамически обновляющееся цифровое значение. Для динамического вывода упорядоченного списка величин $[s=SD(i), q=QD(i), p=PD(i)]$ можно использовать формат `convert([s=S(i), q=Q(i), p=P(i)], string)`. При этом необходимо помнить о том, что в приближенных вычислениях с помощью функции `evalf(S)` Maple по умолчанию выводит 10 значащих цифр. Поэтому до применения конвертирования в строковую переменную необходимо определить формат вывода числовых значений.

1.2. Методы создания анимационного кадра и управление параметрами анимации

Динамическую графику в системе Maple можно создать двумя способами. Первый из них заключается в использовании прямых команд анимации библиотеки `plots`. Их всего 4:

- `animate(f(x,t), x=a..b, t=t0..t1, options)` - анимация плоских кривых, формат использования процедуры практически повторяет формат процедуры двумерной графики `plot`;
- `animate3d(f(x,t), x=a..b, y=c..d, t=t0..t1, options)` - анимация поверхностей, формат использования процедуры практически повторяет формат процедуры трехмерной графики `plot3d`;

- `animatecurve(f(x,t),x=a..b,t=a..b,options)` - анимация пространственных кривых, формат использования процедуры практически повторяет формат процедуры трехмерной кривой `spacecurve`;

– в этих процедурах параметр t играет роль времени. Добавляя в эти процедуры необязательный параметр `frames=N`, мы получаем возможность регулировать число кадров анимации. Возможности указанных программных процедур ограничены. Для создания сложных анимационных моделей необходимо применять второй способ анимации на основе программной процедуры библиотеки `plots display` с необязательной опцией `insequence = true`. Применение этой процедуры без указанной опции создает композицию графических объектов, указанных в теле команды `display`. Поэтому сложные анимационные структуры можно создавать из отдельных графических объектов, собирая их с помощью процедуры в различные структуры, которые, в свою очередь, можно интегрировать функцией `display`. При этом надо заметить, что отдельными графическими объектами, собираемыми процедурой `display` могут быть и анимационные объекты, создаваемые указанными выше способами. Ниже показан простой пример строительства такого графического объекта:

```
> r_minus:=plots[animate](sin(x-t),x=0..2*Pi,t=0..2*Pi,color=red):
r_plus:=plots[animate](sin(x+t),x=0..2*Pi,t=0..2*Pi,color=blue):
plots[display](r_minus,r_plus,insequence=true);
```

При исполнении этой процедуры и запуске анимации в окне Maple возникает бегущая слева-направо красная волна, которая затем сменяется синей, бегущей справа-налево.

Общая идеология построения анимационных моделей представлена на схемах , и подробно разрабатывалась в работах [10], [14], а затем была применена при создании оснащенных динамических моделей нелинейных обобщенно-динамических систем [11–13].

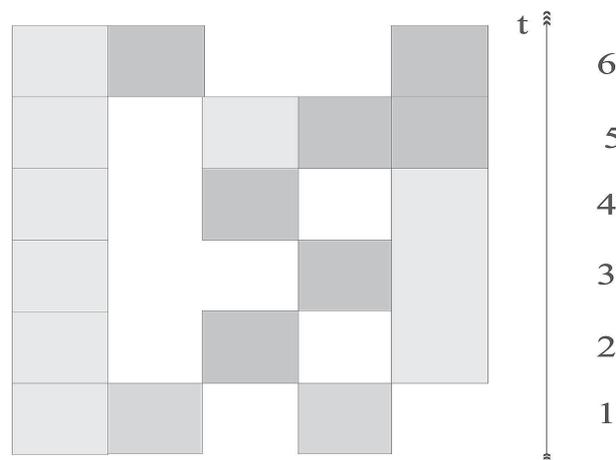


Рис. 2. Схема создания динамической графики. Светлые квадраты - простые графические объекты, темные - сложные, включающие в себя несколько простых, и, возможно, анимированные структуры. Цифры в правой части рисунка - номера кадров анимации. Неизменяющиеся слева квадраты - графическая структура фона – эти элементы должны присутствовать в каждом кадре.

1.3. Пример анимационной процедуры

Для создания достаточно продвинутых многопараметрических программных процедур и удобства их использования необходимо, в свою очередь, создать библиотеку (пакет) программных процедур, а использовать эти процедуры в другом файле с указанием адреса библиотеки процедур в формате

```
>read "library.m":
with(name_library):
```

Синтаксис многопараметрической процедуры `Name_Command(x1,x2,...,xn)` обычно имеет форму:

```
>Library[Name_Command]:=proc(x1,x2,...,xn) local y1,y2,...,ym:
Body of Procedure:
[yk,...,ym]:
end proc:
```

где x_1, x_2, \dots, x_n – параметры команды, y_1, y_2, \dots, y_m – выводимые значения аргументов исполняемой команды, представленные в виде списка (матрицы). Отметим, что сами параметры, как и значения выводимых аргументов, в свою очередь, могут представлять списки или матрицы.

Опишем основные разработанные нами принципы создания многопараметрических программных процедур оснащенной динамической графики на примере создания простой процедуры оснащенной динамической визуализации графика функции $f(x)$ на интервале l :

Как и при создании любой процедуры в Maple программа начинается с преамбулы, в которой перечисляются имена локальных переменных.

Так, например, в процедуре `FuncAnim` оснащенной динамической визуализации графика функции $f(x)$, представляющей фильм из N кадров,

```
FuncAnim:=proc(x,f,l,N,c1,c2,e) local F,X,a,b,i,xx,ff,
FF,f_min,f_max,f_plot,cir,rr,cadr,cadr0:
```

необходимо задать интервал $l \equiv [a, b]$ изменения независимой переменной x :

```
F:=(X)->subs(x=X,f):
a:=l[1]:b:=l[2]:
```

Затем вычисляются значения функции $ff(i)$ в каждой точке $xx(i)$ заданного интервала с шагом $\Delta x = (b - a)/N$, которые выводятся в виде упорядоченной последовательности `FF`:

```
xx:=(i)->evalf(a+(b-a)/N*i,3):
ff:=(i)->evalf(limit(F(X),X=xx(i)),3):
FF:=seq(ff(i),i=0..N):
f_min:=min(FF):f_max:=max(FF):
```

Для графического отображения результатов вычислений используются возможности Maple управления размерами, формами, цветами и другими параметрами графических объектов. Ниже показан фрагмент процедуры, позволяющей отобразить окружность цвета `c2`, диаметра `12pt` с центром в точке $[xx(i), ff(i)]$:

```
cir:=(i)->plots[pointplot]([xx(i),ff(i)],symbol=solidcircle,
symbolsize=14,color=c2):
```

Следующий фрагмент процедуры определяет предел функции в тех точках интервала, в которых функция не определяется явно, причем для корректного графического изображения этого значения добавляется отдельный кадр.

```
if e=0 then
f_plot:=(i)->plot(F(X),X=a..xx(i+1),color=c1,
title=convert([x=xx(i),f=ff(i)],string),
labels=[x,F(x)],titlefont=[TIMES,ROMAN,14],
labelfont=[TIMES,ROMAN,14]):
else
f_plot:=(i)->plot(F(X),X=a..xx(i),color=c1,filled=true,
title=convert([x=xx(i),f=ff(i)],string),
labels=[x,F(x)],titlefont=[TIMES,ROMAN,14],
labelfont=[TIMES,ROMAN,14]):
end if:
cadr0:=plots[pointplot]([xx(0),ff(0)],
symbol=solidcircle,symbolsize=14,color=c2,
title=convert([x=xx(0),f=ff(0)],string),
titlefont=[TIMES,ROMAN,14]):
cadr:=(i)->plots[display](cir(i),f_plot(i)):
plots[display](cadr0,seq(cadr(i),i=1..N),
insequence=true,view=[a..b,f_min..f_max]):
end proc:
```

Созданная таким образом процедура обеспечивает мониторинг процесса изменения значений независимой переменной и соответствующего значения функции на экране. Для вывода этих параметров используется опция `title` команды `plots[pointplot]` в формате `convert([x=xx(0),f=ff(0)],string)`. С помощью этой же функции можно указать и номер соответствующего кадра анимации.

```
FuncAnim(x, sin(x)/x, [0, 2*Pi], 32, red, blue, 2);
```

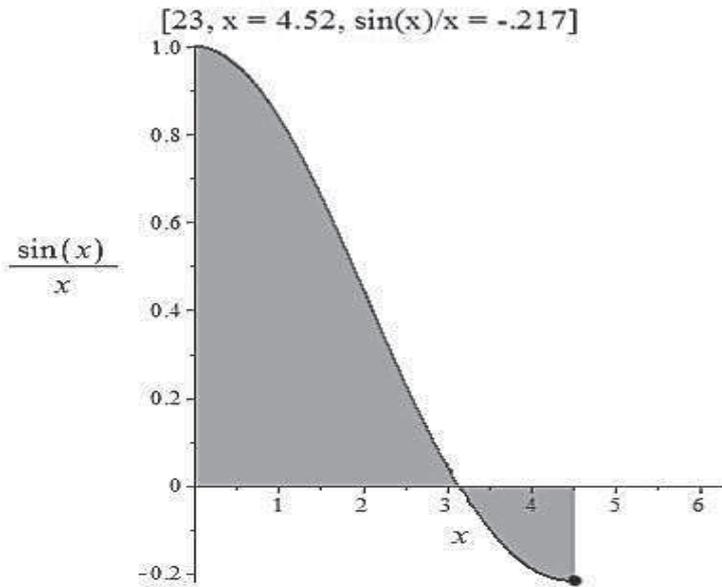


Рис. 3. Исполнение команды `FuncAnim(x, sin(x)/x, [0, 2*Pi], 32, red, blue, 2)` для кадра No 23 из N=32

2. Оснащенная динамическая визуализация адаптированного репера

2.1. Математическая модель дифференциальной геометрии кривых

Как уже было сказано, дифференциальная геометрия кривых основана на применении методов математического анализа функции одной переменной к структуре аналитической геометрии. Основным объектом дифференциальной геометрии кривых является гладкая параметризованная кривая γ в трехмерном евклидовом пространстве E_3 , определяемая относительно декартового репера векторными уравнениями:

$$r = r(t), \dot{r} = \dot{r}(t), \ddot{r} = \ddot{r}(t), \quad (2.1)$$

где r – радиус-вектор точки кривой, заданный упорядоченным набором непрерывных, дважды дифференцируемых декартовых координат точки $x(t), y(t), z(t)$ ⁵. С помощью производных по параметру радиуса вектора, $\dot{r}(t), \ddot{r}(t)$ в каждой точке кривой определяются векторы ортонормированного базиса $\{e\}_\gamma = \{\tau, \nu, \beta\}$ ⁶:

$$\vec{\tau} = \frac{\dot{r}}{|\dot{r}|}; \quad \vec{\beta} = \frac{[\dot{r}, \ddot{r}]}{||[\dot{r}, \ddot{r}]||}; \quad \vec{\nu} = \frac{[\dot{r}, [\dot{r}, \ddot{r}]]}{||[\dot{r}, [\dot{r}, \ddot{r}]]||} \quad (2.2)$$

– единичные векторы касательной, бинормали и нормали, соответственно; $[\vec{a}, \vec{b}]$ – знак векторного произведения векторов \vec{a}, \vec{b} . Присоединение к текущей точке кривой $M(t)$ ортонормированного базиса (2.2) дает нам так называемый адаптированный репер, $\mathcal{R}_t(M(t); \tau(t), \nu(t), \beta(t))$, – правый прямоугольный декартов репер, скользящий вдоль параметризованной кривой и являющийся основным инструментом исследования ее геометрии не только в трехмерном евклидовом, но, вообще

⁵См., например, [5].

⁶Мы придерживаемся классического обозначения векторов адаптированного репера, см., например, [25], [26].

говоря, и в римановом пространстве произвольной размерности. В связи с этим понятие адаптированного репера является чрезвычайно важным для дифференциальной геометрии и требует наглядного представления в курсе дифференциальной геометрии⁷. Далее, для гладкой кривой γ вводится ее натуральный параметр, s , как длина этой кривой:

$$s(t) = \int_{t_0}^t |\dot{r}(t')| dt', \quad (2.3)$$

являющийся монотонно возрастающей функцией параметра t . Разложение производных по натуральному параметру от векторов базиса по векторам же $\{e\}_\gamma$ представляется формулами Френе-Серре:

$$\vec{r}' = k\vec{\nu}; \quad \vec{\nu}' = -k\vec{r} + \varkappa\vec{\beta}; \quad \vec{\beta}' = -\varkappa\vec{\nu}, \quad (2.4)$$

где штрих означает дифференцирование по натуральному параметру а два скаляра:

$$k = k(s); \quad \varkappa = \varkappa(s), \quad (2.5)$$

– кривизна и кручение кривой, соответственно, определяются формулами:

$$k = \frac{|[\dot{\vec{r}}, \ddot{\vec{r}}]|}{|\dot{\vec{r}}|^3}; \quad \varkappa = \frac{(\dot{\vec{r}}, \ddot{\vec{r}}, \ddot{\vec{r}})}{|\dot{\vec{r}}, \ddot{\vec{r}}|^2}; \quad (2.6)$$

где $(\vec{a}, \vec{b}, \vec{c})$ – знак смешанного произведения векторов $\vec{a}, \vec{b}, \vec{c}$. Соотношения (2.5) называются натуральными уравнениями кривой. Теорема о конгруэнтности двух кривых с одинаковыми натуральными уравнениями приводит к тому, что знание в каждой точке кривой ее кривизны и кручения полностью определяет внутреннюю геометрию этой кривой. В дифференциальной геометрии кривых можно различить две задачи, – прямую и обратную. В прямой задаче, которую только обычно и решают в стандартных курсах дифференциальной геометрии, требуется определить натуральные уравнения кривой по заданному ее параметрическому уравнению. В обратной задаче необходимо определить с точностью до движения ее параметрическое уравнение.

2.2. Визуализация натуральных уравнений кривой

Задача визуализации натуральных уравнений произвольной кривой решается является простой – согласно (2.5)–(2.6) для этого необходимо лишь вычислить зависимости (2.5) в каждой точке кривой и изобразить их на совместном графике. В созданном нами пакете программных процедур YuDifGeo эта задача решается с помощью команд GraphicNatural Curvature(Line,param,param0,max_param,cc), YuDifGeo(Line, param,param0,max_param,cc) и GraphicNaturalEquations(Line,param,param0,max_param,cc1,cc2), где Line – параметрические уравнения кривой, задаваемые в списочном виде $[x(t), y(t), z(t)]$; param, param0 – имя параметра кривой и начальное значение параметра, max_param – максимальное значение параметра, cc, cc1, cc2 – цвета изображения кривых. При этом первые две команды изображают кривизну и кручения кривой на отдельном графике, третья команда совмещает эти изображения. Приведем фрагмент программы:

```
>YuDifGeo[GraphicNaturalCurvature]:=
proc(Line,param,param0,max\_param,cc) local param1:
plot([YuDifGeo[CurvatureLine](Line,param,param1),
YuDifGeo[LengthLine](Line,param,param0,param1),
param1=param0..max\_param],
color=cc,labelfont=[TIMES,ROMAN,12],
labels=['s', ''])
end proc:
```

⁷Здесь следует отметить, что в советской системе высшего образования курс дифференциальной геометрии сопровождался большим количеством наглядных учебных пособий. В частности, понятия соприкосновения кривых и адаптированного (натурального) репера кривых демонстрировались на специальных приборах, снабженных подвижной кареткой, скользящей вдоль макета кривой, выполненного из проволоки.

Указанные программные процедуры используют внутренние процедуры пакета – `CurvatureLine`, `TorsionLine` и `LengthLine`, ответственные за вычисление длины дуги кривой, ее кривизны и кручения, соответственно. Продемонстрируем выполнение процедуры изображения натуральных уравнений конической винтовой линии $r = (t \cos(t), t \sin(t), t)$ на отрезке $[0, 4\pi]$, причем график кривизны изображается синим цветом, кручения – красным:

```
>YuDifGeo[GraphicNaturalEquations]([t*cos(t),t*sin(t),t],t,0,4*Pi,blue,red);
```

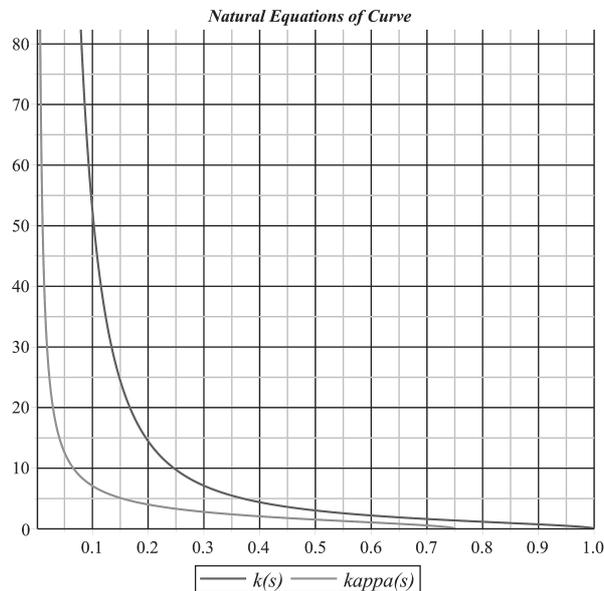


Рис. 4. Исполнение процедуры `GraphicNaturalEquations` для конической винтовой линии.

Аналогичная задача и нахождения натуральных уравнений кривых и их визуализации в системе `Mathematica2` приведена и в цитированной монографии Грэя [6].

2.3. Оснащенная динамическая визуализация адаптированного репера произвольной параметризованной кривой

Перейдем теперь к решению гораздо более сложной задачи – построению оснащенной динамической визуализации адаптированного репера произвольной параметризованной кривой. Отметим, что задача построения изображения адаптированного репера кривой также решается достаточно просто:

```
>YuDifGeo[NaturalReper]:=proc(Line,param,param0,t1,cc1,
cc2,cc3)
local rt1,rt2,rt3,tt,bb,nn,xi,zeta,chi:
tt:=YuDifGeo[NormTangentLine](Line,param,param0):
rt1:=YuDifGeo[ParamLine](subs(param=param0,Line),tt,xi,t1,cc1,scaling=CONSTRAINED):
bb:=YuDifGeo[BinormLine](Line,param,param0):
rt2:=YuDifGeo[ParamLine](subs(param=param0,Line),bb,zeta,t1,cc2,scaling=CONSTRAINED):
nn:=YuDifGeo[MainNormLine](Line,param,param0):
rt3:=YuDifGeo[ParamLine](subs(param=param0,Line),nn,chi,t1,cc3,scaling=CONSTRAINED):
plots[display](rt1,rt2,rt3):end:
```

Как видно, здесь использован ряд внутренних процедур пакета, ответственных за изображение отдельных элементов репера кривой в ее точке, соответствующей значению параметра t_1 .

Значительно более сложно решается задача построения оснащенной динамической модели адаптированного репера кривой. Эта задача решается в созданном пакете [20] программной процедурой

```
YuDifGeo[AnimNatural Reper](Line,param,mini,maxi,c,N,p),
```

где кроме уже введенных параметров $mini$ – минимальное значение параметра t кривой, $maxi$ – максимальное, c – цвет изображения кривой, N – число кадров динамической модели. Параметр p может принимать значение 1 или

любое другое, в том числе и буквенное. При $p=1$ в заголовок рисунка выносятся динамические значения кривизны, кручения и длины дуги. При любом другом значении этого параметра кривизна, кручение и длина дуги кривой отображаются динамическими объемными гистограммами с приложенными к ним динамически обновляемыми значениями указанных скаляров. Для автоматизации процесса построения оптимальной динамической модели необходимо было дать определения ряду графических параметров оснащенной графики, зависящих от конкретного уравнения кривой и заданных границ ее воспроизведения. Для решения этой задачи использовался следующий алгоритм. В каждом кадре определялись минимальные и максимальные значения координат для всех элементов фигуры с помощью встроенной процедуры `min` и `max`, затем составлялся список минимальных и максимальных значений координат фигур по всем кадрам и находились максимальные и минимальные координаты по всей динамической модели. Таким образом, можно было определить максимальные размеры динамического объекта по всем измерениям, что позволило оптимизировать размеры его добавочных элементов. На Рис. 5 показан один кадр исполнения этой программной процедуры для параболической винтовой линии:

$$\vec{r}(t) = (t \cos(t), t \sin(t), 0.015 \cdot t^2) \Rightarrow$$

```
>YuDifGeo [AnimNaturalReper] ([t*cos(t), t*sin(t), 0.015*t^2], t, 0, 16*Pi, red, 24, sss);
```

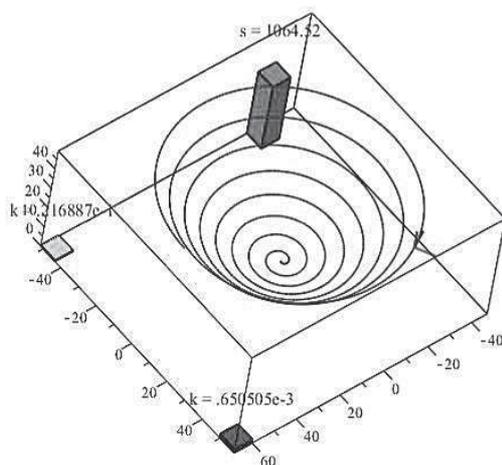


Рис. 5. 23-й кадр исполнения динамической программной процедуры `YuDifGeo [AnimNaturalReper] ([t*cos(t), t*sin(t), 0.015*t^2], t, 0, 16*Pi, red, 24, sss)` для параболической винтовой линии при значении параметра $p=sss$.

Три разноцветных столбика по углам трехмерного динамического изображения отображают динамические значения скаляров s , $k(s)$, $\chi(s)$ — соответственно значению канонического параметра высота этих столбиков изменяется, как изменяются и динамически отображаемые численные значения этих скаляров.

3. Динамическая визуализация восстановления кривой по ее натуральным уравнениям

Как мы отмечали выше, система двух соотношений (2.5) называется *натуральными уравнениями кривой*. Так как натуральное уравнение связывает инварианты кривой, то они не меняются при преобразовании координат или при перемещении кривой относительно этой системы. Основное значение натуральных уравнений состоит в том, что задание их вполне характеризует форму кривой, так что *две кривые с одинаковыми натуральными уравнениями необходимо совпадают по своей форме и могут отличаться только положением пространства, т.е., являются конгруэнтными фигурами*⁸. С другой стороны, кривизна и кручение кривой являются ее двумя локально измеряемыми характеристиками, полностью описывающими свойства кривой: кривизна, $k(s)$,

⁸См., например, [25]

совпадает с абсолютной величиной скорости поворота касательной к кривой, а абсолютная величина кручения, $\varkappa(s)$, – с абсолютной величиной скорости поворота бинормали кривой. Эти две физически измеряемые характеристики кривой являются основой для ориентации на местности и обеспечивают мгновенную привязку к местности мобильных технических средств. Фактическое измерение этих скаляров осуществляется с помощью гиросистем. Математически вопрос привязки перемещаемого по некоторой пространственной кривой $\vec{r} = \vec{r}(t)$ объекта заключается в разрешении натуральных уравнений кривой относительно радиуса-вектора перемещаемой точки. На первый взгляд кажется, что невозможно однозначно восстановить векторную функцию $\vec{r}(t)$ по системе двух натуральных уравнений (2.5). Однако, это возможно сделать при задании начальной точки и начального направления – как раз вследствие указанного свойства конгруэнтности.

Кривизна и кручение кривой, заданной своим параметрическим уравнением, описываются формулами (2.6). Таким образом, задача восстановления кривой (с точностью до пространственного движения) сводится к решению системы двух обыкновенных, существенно нелинейных дифференциальных уравнений:

$$k(s) = \frac{|\dot{\mathbf{r}} \times \ddot{\mathbf{r}}|}{|\dot{\mathbf{r}}|^3}; \quad (3.1)$$

$$\varkappa(s) = \frac{(\dot{\mathbf{r}}, \ddot{\mathbf{r}}, \ddot{\mathbf{r}})}{|\dot{\mathbf{r}} \times \ddot{\mathbf{r}}|^2}. \quad (3.2)$$

с определением связи производных по натуральному параметру s , (длине) и «временем», t :

$$\frac{dt}{ds} = \frac{1}{\dot{\mathbf{r}}}. \quad (3.3)$$

3.1. Задание кривой и вычисление кривизны и кручения по натуральному параметру

Предположим, что нам известны параметрические уравнения кривой в натуральной параметризации, т.е., заданы три функции:

$$x = X(s); y = Y(s); z = Z(s). \quad (3.4)$$

Зададим эту операцию в Maple:

```
restart;
R(s):=[X(s),Y(s),Z(s)];
```

Вычислим производные первого порядка:

```
>tau_1(s):=[diff(X(s),s),diff(Y(s),s),diff(Z(s),s)];
```

Применим теперь следующий прием Ю.Г. Игнатьева [26]. Учтем, что вектор $\vec{\tau}_1$ имеет единичную длину:

$$|\vec{\tau}_1| = 1. \quad (3.5)$$

Вследствие этого как раз и исчезает лишняя степень свободы, поэтому координаты этого вектора можно параметризовать двумя переменными, $\Phi(s)$ и $\Theta(s)$ (аналогичным сферическим координатам):

$$\frac{dX}{ds} = \cos \Phi \cos \Theta; \quad \frac{dY}{ds} = \sin \Phi \cos \Theta; \quad \frac{dZ}{ds} = \sin \Theta : \quad (3.6)$$

```
diff(X(s),s) = cos(Phi(s))*cos(Theta(s));
diff(Y(s),s) = sin(Phi(s))*cos(Theta(s));
diff(Z(s),s) = sin(Theta(s));
```

где $\Phi(s)$ и $\Theta(s)$ – произвольные пока функции натурального параметра.

Сделаем указанные подстановки и вычислим затем производную радиуса вектора в новых переменных:

```
>DR:={diff(X(s),s)=cos(Phi(s))*cos(Theta(s)),diff(Y(s),s)
=sin(Phi(s))*cos(Theta(s)),diff(Z(s),s)=sin(Theta(s))};
```

Запишем производную радиуса вектора, $\vec{r}'(s)$, с помощью новых переменных:

```
>tau(s):=subs(DR,tau_1(s));
u(s):=tau(s)[1];v(s):=tau(s)[2];w(s):=tau(s)[3];
tau(s):=[cos(Phi(s))*cos(Theta(s)),
sin(Phi(s))*cos(Theta(s)),sin(Theta(s))];
```

Вычислим теперь производные второго порядка:

```
R2(s):=[diff(u(s),s),diff(v(s),s),diff(w(s),s)];
R2(s):=[-sin(Phi(s))*diff(Phi(s),s)*cos(Theta(s))-
cos(Phi(s))*sin(Theta(s))*diff(Theta(s),s),
cos(Phi(s))*diff(Phi(s),s)*cos(Theta(s))-
sin(Phi(s))*sin(Theta(s))*diff(Theta(s),s),
cos(Theta(s))*diff(Theta(s),s)];
```

и, наконец, производные третьего порядка:

```
>R3(s):=[diff(u(s),s$2),diff(v(s),s$2),diff(w(s),s$2)];
R3(s):=[-cos(Phi(s))*diff(Phi(s),s)^2*cos(Theta(s))-
sin(Phi(s))*diff(Phi(s),s,2)*cos(Theta(s))+
2*sin(Phi(s))*diff(Phi(s),s)*sin(Theta(s))*
diff(Theta(s),s)-cos(Phi(s))*cos(Theta(s))*
diff(Theta(s),s)^2-cos(Phi(s))*
sin(Theta(s))*diff(Theta(s),s,2),
-sin(Phi(s))*diff(Phi(s),s)^2*cos(Theta(s))+cos(Phi(s))*
diff(Phi(s),s,2)*cos(Theta(s))-2*cos(Phi(s))*diff(Phi(s),s)*
sin(Theta(s))*diff(Theta(s),s)-sin(Phi(s))*cos(Theta(s))*
diff(Theta(s),s)^2-sin(Phi(s))*s
in(Theta(s))*diff(Theta(s),s,2),
-sin(Theta(s))*diff(Theta(s),s)^2+
cos(Theta(s))*diff(Theta(s),s,2)];
```

3.2. Вычисление кривизны и кручения кривой

Предполагая, теперь, что параметрические уравнения кривой Γ известны, вычислим ее кривизну и кручение по стандартным формулам дифференциальной геометрии (3.1), (3.2). При этом скалярное и векторное произведения векторов \vec{a} и \vec{b} будем вычислять с помощью команд `innerprod(a,b)` и `crossprod(a,b)` библиотеки `linalg`, соответственно:

```
>k(s):=sqrt(simplify(linalg[innerprod](R2(s),R2(s))));
k(s):=(diff(Phi(s),s)^2*cos(Theta(s))^2+diff(Theta(s),s)^2)^(1/2);
```

С помощью команды `scalarmul(a,alpha)` библиотеки `linalg` произведем умножение вектора $\vec{r}''(s)$ на число α – тем самым получаем единичный вектор \vec{v} сопровождающего трехгранника кривой:

```
>nu(s):=linalg[scalarmul](R2(s),1/k(s));
nu(s):=vector([1/(diff(Phi(s),s)^2*cos(Theta(s))^2+
diff(Theta(s),s)^2)^(1/2)*
(-sin(Phi(s))*diff(Phi(s),s)*cos(Theta(s))-
cos(Phi(s))*sin(Theta(s))*diff(Theta(s),s)),
1/(diff(Phi(s),s)^2*cos(Theta(s))^2+diff(Theta(s),s)^2)^(1/2)*
(cos(Phi(s))*diff(Phi(s),s)*cos(Theta(s))-
sin(Phi(s))*sin(Theta(s))*diff(Theta(s),s)),
1/(diff(Phi(s),s)^2*cos(Theta(s))^2+
diff(Theta(s),s)^2)^(1/2)*cos(Theta(s))*diff(Theta(s),s)]);
```

Убедимся в том, что первая и вторая производные радиуса-вектора ортогональны (так как вторая производная есть производная от единичного вектора \vec{v}).

```
simplify(linalg[innerprod](tau(s),nu(s)));
```

Вычислим векторное произведение этих же векторов:

```
V_DR_D2R(s):=linalg[crossprod](tau(s),R2(s));
V_DR_D2R(s) :=vector([sin(Phi(s))*cos(Theta(s))^2*diff(Theta(s),s)-
sin(Theta(s))*(cos(Phi(s))*diff(Phi(s),s)*cos(Theta(s))-
sin(Phi(s))*sin(Theta(s))*diff(Theta(s),s)),
sin(Theta(s))*(-sin(Phi(s))*diff(Phi(s),s)*
cos(Theta(s))-cos(Phi(s))*sin(Theta(s))*diff(Theta(s),s))-
cos(Phi(s))*cos(Theta(s))^2*diff(Theta(s),s),
cos(Phi(s))*cos(Theta(s))*(cos(Phi(s))*diff(Phi(s),s)*
cos(Theta(s))-sin(Phi(s))*sin(Theta(s))*diff(Theta(s),s))-
sin(Phi(s))*cos(Theta(s))*(-sin(Phi(s))*diff(Phi(s),s)*
cos(Theta(s))-cos(Phi(s))*sin(Theta(s))*diff(Theta(s),s))]);
```

С помощью скалярного умножения полученного результата и вектора третьей производной получим кручение кривой:

```
kappa(s):=simplify(linalg[innerprod](V_DR_D2R(s),R3(s))/k(s)^2);
kappa(s) :=(sin(Theta(s))*diff(Phi(s),s)^3*cos(Theta(s))^2+
2*sin(Theta(s))*diff(Theta(s),s)^2*diff(Phi(s),s)-
diff(Theta(s),s)*diff(Phi(s),s^2)*cos
(Theta(s))+diff(Phi(s),s)*cos(Theta(s))*diff(Theta(s),s^2)/(diff
(Phi(s),s)^2*cos(Theta(s))^2+diff(Theta(s),s)^2);
%
kappa(s) :=(diff(Phi(s),s)*cos(Theta(s))*diff(Theta(s),s^2)-
diff(Theta(s),s)*diff(Phi(s),s^2)*cos(Theta(s))+
2*sin(Theta(s))*diff(Theta(s),s)^2*diff(Phi(s),s)+
sin(Theta(s))*diff(Phi(s),s)^3*
cos(Theta(s))^2)/(diff(Phi(s),s)^2*cos(Theta(s))^2+
diff(Theta(s),s)^2);
```

3.3. Формирование системы дифференциальных уравнений

Напомним, что натуральные уравнения кривой — это пара дифференциальных уравнений (2.5), в которых $k(s)$ и $\varkappa(s)$ в правых частях заданы как функции натурального параметра s . Правые части этих уравнений мы будем задавать, а левые части этих уравнений определяются дифференциальными выражениями, которые мы получили выше. Таким образом, мы получаем систему двух обыкновенных дифференциальных уравнений второго порядка относительно двух неизвестных функций — $\Phi(s)$ и $\Theta(s)$. Поскольку эти уравнения существенно нелинейные, их необходимо решать численными методами.

Для численного интегрирования этой системы необходимо привести ее к нормальному виду, т.е., к системе ОДУ 1-го порядка, разрешенных относительно производных. Для этого обозначим первые производные от искомых функций с помощью новых функций, $\Phi_S(s)$ и $\Theta_S(s)$:

$$\frac{d\Phi(s)}{ds} = \Phi_S(s); \quad (3.7)$$

$$\frac{d\Theta(s)}{ds} = \Theta_S(s); \quad (3.8)$$

```
D1:=diff(Phi(s),s)=Phi[S](s);D2:=diff(Theta(s),s)=Theta[S](s);
```

```
Система ОДУ решается с помощью команды
>dsolve(система ОДУ union Начальные условия, type=numeric,
method=classical, output=listprocedure):
```

— при этом получаем численные решения классическим методом, который является комбинацией методов Эйлера, Хейца, Рунге - Куддта и Адамса - Бэшфорда, причем решения выводятся в виде списка. Как показывают вычисления, этот комбинированный метод дает лучшие результаты, чем стандартный метод Рунге - Куддта в случаях, когда кривая имеет много изгибов.

```

> Sol[1]:= dsolve(System1 union Inits,
{X(s),Y(s),Z(s),Phi(s),Theta(s),Phi[S](s),Theta[S](s)},
type=numeric, method=classical,output=listprocedure);
Sol[2]:= dsolve(System2 union Inits, {X(s),Y(s),Z(s),
Phi(s),Theta(s),Phi[S](s),Theta[S](s)},
type=numeric,method=classical, output=listprocedure);
Sol[3]:= dsolve(System3 union Inits,
{X(s),Y(s),Z(s),Phi(s),Theta(s),Phi[S](s),Theta[S](s)},
type=numeric,method=classical, output=listprocedure);
Sol[4]:= dsolve(System4 union Inits, {X(s),Y(s),Z(s),
Phi(s),Theta(s),Phi[S](s),Theta[S](s)},
type=numeric,method=classical, output=listprocedure);

```

3.4. Компьютерное моделирование натуральных уравнений кривой

Подставляя теперь в правую часть натуральных уравнений кривой полученные выражения для кривизны и кручения, а в левые – заданные функции параметра s , получим систему двух нелинейных обыкновенных дифференциальных уравнений 3-го порядка относительно двух углов $\Phi(s)$ и $\Theta(s)$. Как мы отмечали выше, точка начала кривой и ее начальное направление могут быть заданы произвольно. Соответственно этому произволу зададим начальные условия в виде:

```

>InitConditions:={X(0)=0,Y(0)=0,Z(0)=0,Phi(0)=0,
Theta(0)=0,Phi[S](0)=1,Theta[S](0)=0};

```

Зададим теперь конкретные натуральные уравнения кривой, т.е., конкретизируем функции $k(s)$ и $\varkappa(s)$. Рассмотрим несколько примеров. На последующих рисунках представлены результаты компьютерного восстановления кривых по их натуральным уравнениям.

3.5. Кривая 1:

$$k(s) = 1, \varkappa(s) = 0. \quad (3.9)$$

Эта кривая, как понятно из геометрического смысла кривизны и кручения, является единичной окружностью. Одновременно – это хороший пример для проверки работы комплекса программ.

```

>p1_xyz:=(n)->
[seq([x_1(i/10),y_1(i/10),z_1(i/10)],i=0..n)]:
g1_xyz:=(n)->
plots[spacecurve](p1_xyz(n),color=black,thickness=2,
axes=FRAME,titlefont=[TIMES,BOLD,14],
labelfont=[TIMES,BOLD,14],labels=[y,x,z]):
g1_xyz(160);

```

3.6. Кривая 2:

$$k(s) = \frac{s^2}{1+s^4}; \varkappa(s) = -\frac{s^2}{1+s^2}. \quad (3.10)$$

```

>p2_xyz:=(n)->
[seq([x_2(i/10),y_2(i/10),z_2(i/10)],i=0..n)]:
>g2_xyz:=(n)->plots[spacecurve](p2_xyz(n),color=black,
thickness=2,axes=FRAME,titlefont=[TIMES,BOLD,14],
labelfont=[TIMES,BOLD,14],labels=[y,x,z]):
g2_xyz(160);

```

$$k(s) = \frac{s^2}{1+s^4}; \varkappa(s) = -\frac{s^2}{1+s^2}. \quad (3.11)$$

```

>p2_xyz:=(n)->
[seq([x_2(i/10),y_2(i/10),z_2(i/10)],i=0..n)]:
>g2_xyz:=(n)->plots[spacecurve](p2_xyz(n),color=black,
thickness=2,axes=FRAME,titlefont=[TIMES,BOLD,14],
labelfont=[TIMES,BOLD,14],labels=[y,x,z]):
g2_xyz(160);

```

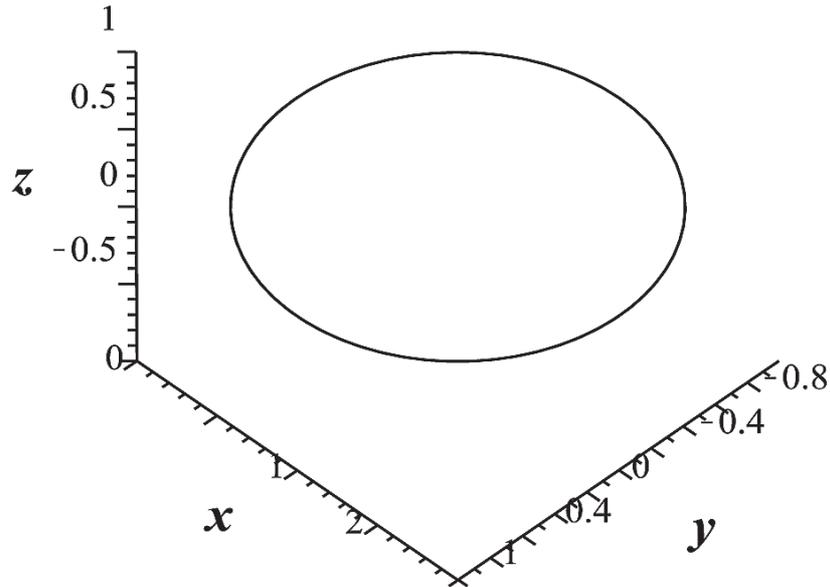


Рис. 6. Восстановление кривой 1 по ее натуральным уравнениям (3.9). Кривая Γ - единичная окружность в пространстве, как и следует из теории [25].

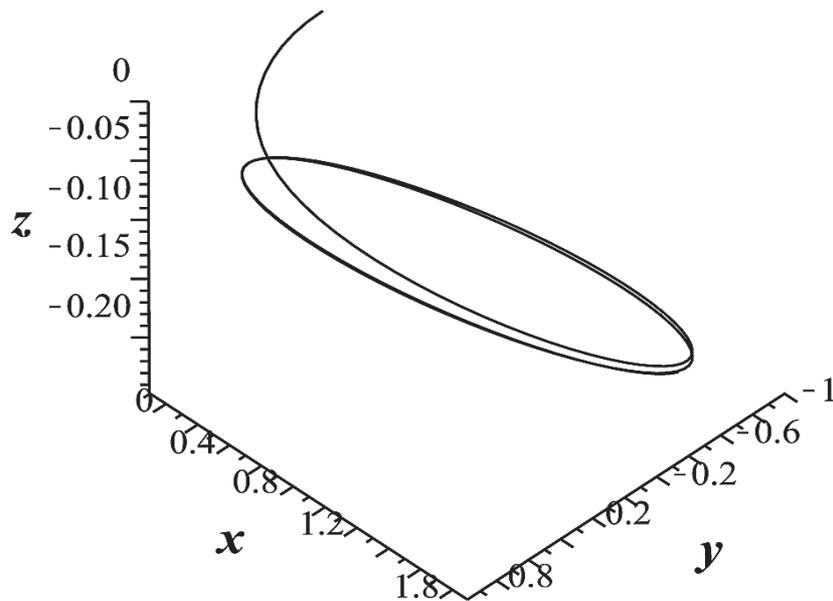


Рис. 7. Восстановление кривой 2 по ее натуральным уравнениям (3.11). Кривая Γ - искаженная винтовая линия.

Из рисунка видно, что эту кривую можно охарактеризовать как деформированную винтовую линию с переменным шагом и наклонной осью.

3.7. Кривая 3:

$$k(s) = \cos s; \quad \varkappa(s) = \sin s. \quad (3.12)$$

```
>p3_xyz:=(n)->
[seq([x_3(Pi*i/100),y_3(Pi*i/100),z_3(Pi*i/100)],
i=0..n)]:
g3_xyz:=(n)->plots[spacecurve](p3_xyz(n),color=black,
thickness=2,axes=FRAME,
titlefont=[TIMES,BOLD,14],
labelfont=[TIMES,BOLD,14],labels=[y,x,z]):
g3_xyz(1000);
```

Натуральные уравнения этой кривой приведены в книге А.П. Нордена [25].

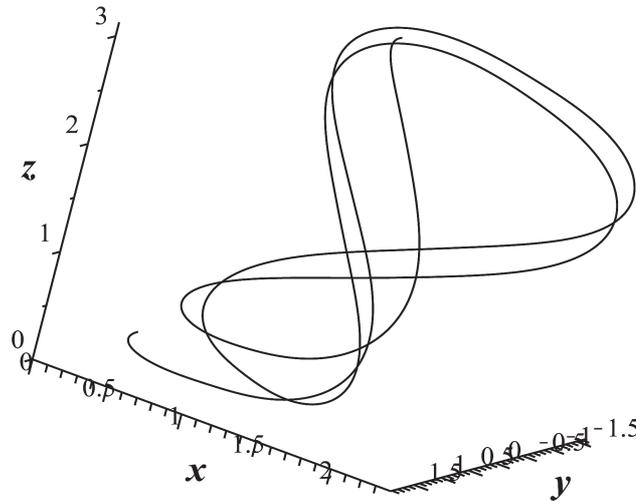


Рис. 8. Восстановление кривой 3 по ее натуральным уравнениям (3.12).

Для получения анимационных моделей восстановленных кривых достаточно применить процедуру `plots[display](seq(g3(i)), i=1..N, insequence=true)`, где N – число кадров трехмерной анимации.

4. Динамическая визуализация геодезических линий

4.1. Определение геодезических линий

Пусть $g_{ik}(s)$ – метрический тензор риманова пространства

$$ds^2 = g_{ik} dx^i dx^k \quad i, k = \overline{1, n}, \quad (4.1)$$

и параметрические уравнения линии имеют вид:

$$\vec{r} = \vec{r}(s),$$

где s – канонический параметр (длина дуги линии).

Метрический тензор $g_{ik}(x)$, в свою очередь, определяет символы Кристоффеля первого и второго рода

$$\Gamma_{i,jk} = \frac{1}{2} \left(\frac{\partial g_{ij}}{\partial x^k} + \frac{\partial g_{ki}}{\partial x^j} - \frac{\partial g_{jk}}{\partial x^i} \right), \quad \Gamma_{jk}^l = g^{il} \Gamma_{i,jk}.$$

Тогда геодезические линии определяются системой n обыкновенных дифференциальных уравнений второго порядка

$$\frac{d^2 x^i}{ds^2} + \Gamma_{jk}^i \frac{dx^j}{ds} \frac{dx^k}{ds} = 0. \quad (4.2)$$

Как известно⁹, строгим дифференциальным следствием уравнений (4.2) является соотношение нормировки:

$$g_{ik} \frac{dx^i}{ds} \frac{dx^k}{ds} = \text{Const} \Rightarrow g_{ik} \frac{dx^i}{ds} \frac{dx^k}{ds} = 1. \quad (4.3)$$

Поставим задачу нахождения геодезических произвольных римановых пространств. При этом, учитывая соотношение нормировки касательного вектора геодезической $u^i \equiv dx^i/ds$ достаточно проинтегрировать $(n-1)$ уравнение системы (4.2).

⁹См., например, [24]

Для решения системы обыкновенных дифференциальных уравнений необходимо задать начальные условия, задавая координаты начальной точки и координаты направляющего вектора:

$$M(x_0) = M_0 \Rightarrow x^i(0) = x_0^i; \quad (4.4)$$

$$\left. \frac{dx^i}{ds} \right|_{s=0} = u_0^i. \quad (4.5)$$

Однако, вследствие соотношения нормировки (4.3) произвольными являются лишь $n - 1$ координата направляющего вектора.

Опишем библиотеку программных процедур «Geodesic_lines» создания оснащенных динамических графиков геодезических, предназначенную для проведения исследований в области римановой геометрии средствами СКМ Maple¹⁰. Библиотека позволяет решить следующие задачи:

1. Нахождение уравнений геодезических линий поверхности;
2. Приведение уравнений геодезических к нормальной системе обыкновенных дифференциальных уравнений (ОДУ);
3. Численное решение системы ОДУ;
4. Графическое представление решения по сгенерированным начальным условиям;
5. Динамическая визуализация движения частицы в гравитационном поле.

Разработанный программный комплекс библиотеки «Geodesic_lines» предназначен для автоматизированного нахождения и построения геодезических линий и геодезических трубок в трехмерных (локально евклидовых) и четырехмерных (псевдоевклидовых) римановых пространствах и их оснащенной динамической визуализацией с автоматической оптимизацией графических параметров в системе компьютерной математики Maple.

При этом средствами СКМ Maple:

1. Формируется система дифференциальных уравнений математической модели
2. Анализируется полученная система уравнений
3. Формируется система начальных условий
4. Создается процедура численного интегрирования полученной системы дифференциальных уравнений с группами начальных условий
5. Создаются анимационные модели динамической системы

Программный комплекс оптимизирует графические параметры, а также согласовывает положение геодезической линии с визуализированной частью поверхности.

Программный комплекс отличается от известных аналогов, во-первых, простотой ввода программ, которые задаются одной командой с помощью задания метрики пространства и начальных условий, полной автоматизацией, а также возможностью исследования геодезических в произвольных трехмерных и четырехмерных римановых пространствах и возможностью оснащенной динамической визуализации геодезических линий и их трубок. Исследование движения геодезической трубки позволяет извлекать дополнительную информацию о свойствах трехмерного и четырехмерного риманова пространства.

4.2. Библиотека общих программных процедур

Средствами пакета Maple задаются параметрические уравнения поверхности, формируется соответствующий им метрический тензор и находятся уравнения геодезических линий.

Для получения уравнений геодезических можно использовать команду `geodesic_egns(coord,param,Cf2)` встроенной библиотеки “tensor”, где `coord` – список имен координат, `param` – параметр кривой, `Cf2` – символы Кристоффеля второго рода. При этом уравнения геодезических выводятся в произвольном порядке.

¹⁰Эти исследования могут носить, как учебный, так и научный характер.

Для вывода упорядоченной системы уравнений геодезических используем авторскую методику, входящую в библиотеку процедур «Geodesic_lines» [21–23].

```
> Geodesic_lines[EqsGeodesic] := proc (Coord, S, P)
> GG := array(1..n, 1..n, symmetric, sparse):
> g := create([-1, -1], eval(GG)):
> Cf2 := Christoffel2(g_1, Cf1):
> Geod := [seq(diff(Coord[i], S) + dd[i] = 0, i = 1..n)]:
```

Получаем систему дифференциальных уравнений первого порядка, разрешенных относительно производных.

Преобразуем эти уравнения в нормальную систему дифференциальных уравнений относительно шести переменных $x1(s), x2(s), x3(s), v1(s), v2(s), v3(s)$ с помощью процедуры созданной библиотеки:

```
> Geodesic_lines[EqsGeodesicNorm] := proc (Eqs, Coord, NewCoord, S)
> Eqs1 := [seq(diff(XX[i], S) = UU(S)[i], i = 1..n)]:
> Eqs2 := subs({seq(Coord[i] = Coord[i](S), i = 1..n)}, (GEO, [diff(UU(S)[i], S), i = 1..n])):
> Geod_eqs := [Eqs1, op(Eqs2)]: Geod_eqs:
```

Для правильного задания начальных условий задачи вводим условие нормировки в виде алгебраического уравнения второго порядка относительно переменных $v1(s), v2(s), v3(s)$.

Численное интегрирование полученной системы дифференциальных уравнений осуществляется с помощью функции `dsolve(ODE, ICs, options)`. Здесь ODE – система дифференциальных уравнений с указанием начальных условий, ICs – выражение, задающее начальные условия.

```
> Geodesic_lines[EqsGeodesicSolve] := proc (Eqsn, Coord, NewCoord, P, S, Inits)
> VV := Geodesic_lines[quard_vector](GG, Inits[2], Coord, Inits[1]):
> V0 := linalg[scalarmul](Inits[2], 1/sqrt(VV)):
> Inits0 := {seq(Coord[i] = Inits[1][i], i = 1..n), seq(NewCoord[i] = V0[i], i = 1..n)}:
> SS := dsolve({Eqsn[1], Eqsn[2]} union Inits0, type = numeric, output = listprocedure):
> end proc:
```

По заданным начальным условиям процедура возвращает особый тип данных, позволяющих найти решения в любой точке и построить график решений.

Разработанная библиотека процедур дает возможность, вводя параметрические уравнения различных поверхностей, проводить исследование геодезических на 2-мерных поверхностях, в трехмерных и четырехмерных евклидовых и псевдоевклидовых римановых пространствах, в частности движение частиц в различных гравитационных полях.

Ниже приведены примеры построения геодезических и их анимации, представленные в формате 3D – графиков.

Построим геодезическую на однополостном гиперboloиде вращения:

$$\vec{r} = (2 \cosh(u/2) \cos(v), 2 \cosh(u/2) \sin(v), u).$$

```
E := Geodesic_lines[Geodesic] ([ [u, v], [-5, 5], [-2*Pi, 2*Pi] ],
s, 20, [2*cosh(u/2)*cos(v), 2*cosh(u/2)*sin(v), u],
[U, V], [[0, 1], [Pi/32, 1]], 200, 0, animate):
```

Тогда применение опции E[1] выписывает систему уравнений геодезических в стандартном виде:

$$\left[\frac{d^2}{ds^2} u(s) - 1/2 \frac{\sinh(1/2 u) \left(-\left(\frac{d}{ds} u(s)\right)^2 + 4 \left(\frac{d}{ds} v(s)\right)^2 \right)}{\cosh(1/2 u)} = 0, \right. \\ \left. \frac{d^2}{ds^2} v(s) + \frac{\sinh(1/2 u) \left(\frac{d}{ds} v(s)\right) \frac{d}{ds} u(s)}{\cosh(1/2 u)} = 0 \right]$$

а применение опции E[2] выписывает систему уравнений геодезических в нормальном виде:

$$\left[\left[\frac{d}{ds} u(s) = U(s), \frac{d}{ds} v(s) = V(s) \right], \right.$$

$$\left. \left[\frac{d}{ds} U(s) = -1/2 \frac{\sinh(1/2 u(s)) \left((U(s))^2 - 4(V(s))^2 \right)}{\cosh(1/2 u(s))}, \right. \right.$$

$$\left. \left. \frac{d}{ds} V(s) = -\frac{\sinh(1/2 u(s)) V(s) U(s)}{\cosh(1/2 u(s))} \right] \right]$$

Применение опции E[3] (5) дает численное решение системы уравнений геодезических относительно координат (u, v) при $s = 5$:

$$[.573616362991073991, 3.45798989479676732],$$

а применение опции E[4] дает анимированный график геодезической на гиперboloиде (см. Рис. 9).

4.3. Динамическая визуализация оптических лучей в неоднородной среде

Пусть $n_{ik}(\vec{r})$ – тензор преломления оптически-прозрачной среды. Пусть параметрические уравнения движения фотона имеют вид: $\vec{r} = \vec{r}(s)$, где s – параметр.

Бесконечно малая оптическая длина пути фотона $d\tau$ в анизотропной оптической среде с тензором преломления n_{ik} определяется формулой $d\tau^2 = n_{ik} dx^i dx^k$.

Абсолютная величина скорости фотона v в среде в направлении $d\vec{r}$ определяется соотношением: $v = \sqrt{n_{ik} \dot{x}^i \dot{x}^k}$, где $\dot{x}^i = \frac{dx^i}{ds}$.

Функция Лагранжа светового луча в оптически прозрачной неоднородной среде совпадает с абсолютной скоростью фотона в среде: $L = v(s) = \sqrt{n_{ik} \dot{x}^i \dot{x}^k}$.

Тогда уравнения Эйлера имеют вид:

$$\frac{d^2 x^i}{d\tau^2} + \Omega_{jk}^i \frac{dx^j}{d\tau} \frac{dx^k}{d\tau} = 0.$$

Здесь введены компоненты тензора $\Omega_{jk}^i(\vec{r})$ валентности 3 дважды ковариантного и единожды контрвариантного:

$$\Omega_{i,jk} = \frac{1}{2} \left(\frac{\partial n_{ij}}{\partial x^k} + \frac{\partial n_{ki}}{\partial x^j} - \frac{\partial n_{jk}}{\partial x^i} \right), \quad \Omega_{jk}^l = n^{il} \Omega_{i,jk},$$

n^{il} – контрвариантный тензор преломления. Тензоры $\Omega_{ij,k}, \Omega_{k,j}^i$ являются симметричными по паре индексов: $\Omega_{ij,k} = \Omega_{ji,k}$ и $\Omega_{kj}^i = \Omega_{jk}^i$.

Таким образом, уравнения движения фотонов в оптически прозрачной среде с тензором преломления n_{ik} формально ничем не отличаются от уравнений геодезических в метрике n_{ik} . При этом квадрат скорости фотона в масштабе оптической длины равен единице:

$$n_{ik} \frac{dx^i}{d\tau} \frac{dx^k}{d\tau} \equiv 1.$$

Тогда математическая модель геометрической оптики характеризуется следующим утверждением: в пределе геометрической оптики траектория фотона (луч света) в оптически прозрачной анизотропной и неоднородной среде с тензором преломления $n_{ik}(x)$ совпадает с геодезической в римановом пространстве с метрическим тензором $g_{ik}(x) = n_{ik}(x)$.

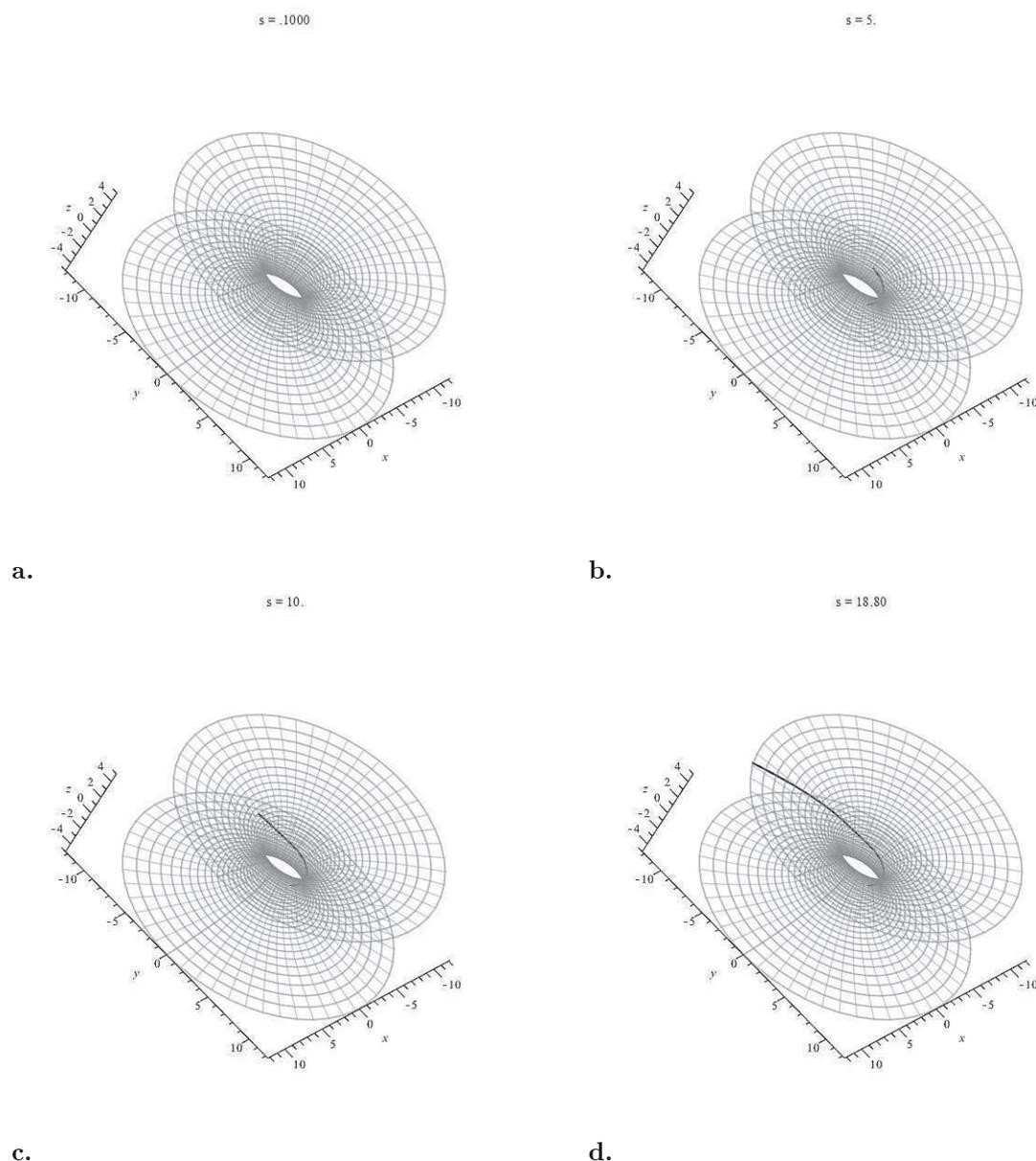


Рис. 9. Анимация геодезических на однополостном гиперboloиде вращения. Вверху графиков находится динамическое табло с моментальным значением канонического параметра: а – Кадр 0 ($s=0.1$), б – Кадр 50 ($s=5$), в – Кадр 100 ($s=10$), д – Кадр 188 ($s=18.8$).

Этим утверждением устанавливается тесная связь между геометрической оптикой и римановой геометрией, а указанное взаимно - однозначное соответствие позволяет использовать авторскую библиотеку процедур «Geodesic_lines» для построения модели движения пучка траекторий фотонов в анизотропной оптической среде, исследовать основные геометрические свойства этого пучка и представить анимацию этого движения. В дальнейшем пучок геодезических, характеризующихся заданием начальных условий на определенной гиперповерхности, будем называть *геодезической трубкой*, как и пучок световых лучей – *световой трубкой*.

Процедура `geo_eqns_solv(Coord,S,G,M0,V0,a)` предназначена для получения упорядоченной системы уравнений геодезических и основанного на этой системе численного решения и графического представления решения этой системы как для анизотропных оптических сред, так и для релятивистских полей тяготения. В качестве параметров процедуры выступают: `Coord` - список координат (цилиндрические, сферические и т.д.), `S` - параметр, `G` - метрика, `M0` - координаты центральной точки M_0 геодезической трубки, `K0` - координаты начального направляющего вектора K_0 трубки, `a` - начальный радиус трубки геодезических линий (лучей).

Средствами пакета Maple задаются координаты матрицы тензора преломления (метрического тензора), на основе матрицы формируется соответствующий тензор преломления, вычисляются контрвариантный тензор преломления и символы Кристоффеля первого и второго рода.

Для получения уравнений световых лучей вычисляются контрвариантные векторы скорости частицы u^i и ее ускорения $du \equiv d/ds u(s)^i$:

```
> Geodesic_lines[OpticEqsGeodesic]:=proc(Coord,S,P)
> u:=create([1],array(1..nn,U(S)));
> du:=create([1],array(1..nn,dU(S)));
> v:=create([1],array(1..nn,[seq(v[i](S),i=1..nn)]));
> end proc:
```

С помощью процедуры нахождения внутреннего произведения тензоров по i -ой и k -той координатам находится свертка: $dC = \Gamma_{i,jk} u^j u^k$ для получения уравнения геодезических.

С учетом соотношения нормировки получается система ОДУ 2-го порядка, которая приводится к виду нормальной упорядоченной системы обыкновенных дифференциальных уравнений применительно к заданному тензору преломления относительно шести переменных:

```
x1(s), x2(s), x3(s), v1(s), v2(s), v3(s):
```

```
> Geod1:= [seq(diff(X(S)[i],S)=v[i],i=1..nn)]:
> Geod2:= [seq(diff(v[i](S),S)=dd[i],i=1..nn)]:
```

Начальные условия определяются: координатами центральной точки M_0 геодезической трубки и координатами направляющего вектора K_0 геодезической трубки.

Процедура авторской библиотеки позволяет сгенерировать группу начальных условий, используя некоторый начальный радиус a трубки геодезических линий (лучей). При этом определяются координаты точек по начальному сечению геодезической трубки и координаты ортов в каждой точке, коллинеарных заданному направлению K_0 :

```
> kk:=(i)->eval(subs(M(i),KK)):
> V:=(i)->scalarmul(K,\1/kk(i)):
> VV:=(i)->subs(K0,v(i)):
```

Создадим группы процедур численного решения систем уравнений геодезических по сгенерированным начальным условиям. Изобразим график геодезических трубок по сгенерированным начальным условиям. Визуализация геодезической линии или геодезической трубки в трехмерном пространстве может осуществляться как в статистическом 3-х мерном формате, так и в виде 3-х мерной оснащенной динамической графики. При этом геодезические линии трубки автоматически окрашиваются в разные цвета, что облегчает анализ модели.

Рассмотрим пример построения световой трубки в неоднородной анизотропной оптической среде с тензором преломления:

$$\|n_{ik}\| = \begin{pmatrix} \frac{2 \sin(x1) + \cos(x1)}{\cos(x1)^2} & 0 & 0 \\ 0 & 1 - \cos(x1)^2 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (4.6)$$

Введем тензор преломления с помощью имени `gg2` и выведем с помощью процедуры `optica` световую трубку с началом в точке $M_0 = (0, -1, 0)$, направлении $\vec{u}_0 = (0.1, 0, 0)$ в графическом формате (Рис. 10).

```
gg2:= [[(2*sin(x1)+cos(x1))/cos(x1)^2,0,0],[0,1-cos(x1)^2,0],[0,0,1]]:
VV:=optica([x1,x2,x3],s,50,gg2,[0.1,0,0],[0,-1,0],50,1,graphic):
VV;
```

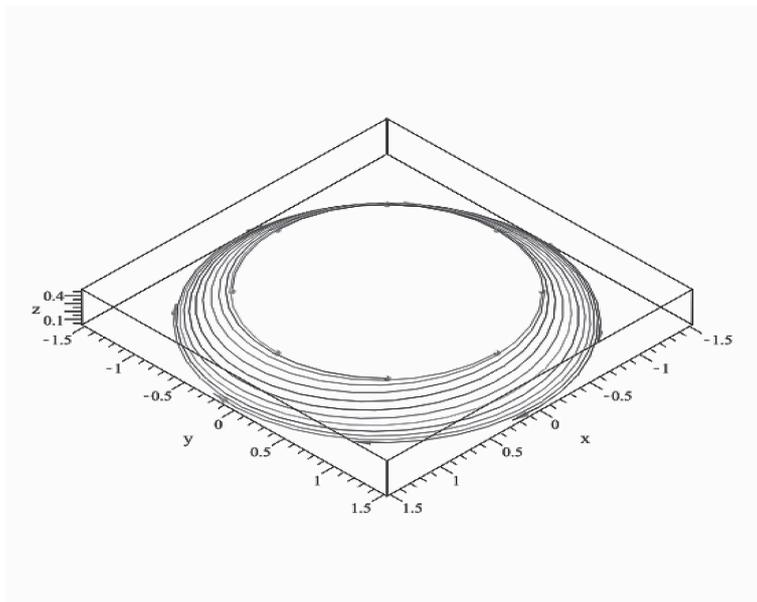


Рис. 10. Световая трубка в оптической среде с тензором преломления (4.6).

Рассмотрим еще один пример построения оптических трубок в среде с неоднородным и анизотропным тензором преломления:

$$\|n_{ik}\| = \begin{pmatrix} \frac{1}{1+x^2} & 0 & 0 \\ 0 & \frac{\cos(x)^2}{1+x^2} & 0 \\ 0 & 0 & x^2 \end{pmatrix}. \quad (4.7)$$

Сгенерируем этот тензор и световую трубку (Рис. 11). На этом рисунке четко видны плоскость поворота световых лучей и фокус в вершине.

```
gg4:=[[1/(1+x^2),0,0],[0,cos(x)^2/(1+x^2),0],[0,0,x^2]]:
AA:=optica([x1,x2,x3],s,7,gg4,[0.1,0,0],
[0,-1,0],[[-1.6,1.6],[-1.6,1.6],[0,1]],50,1,graphic):
```

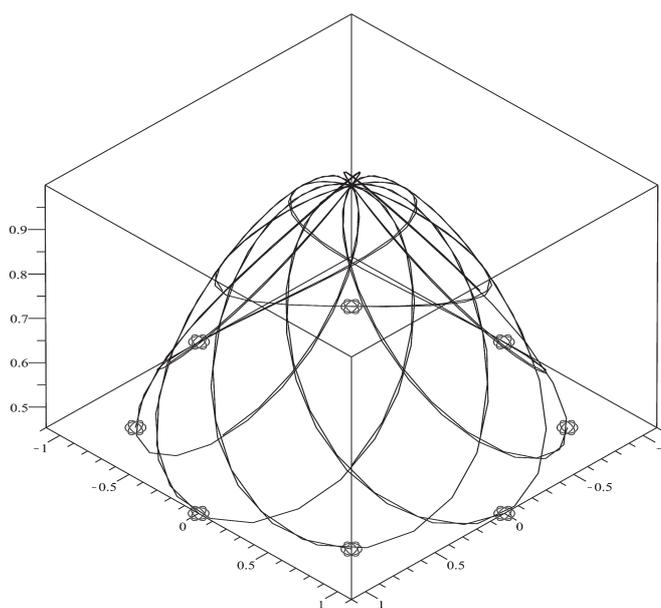


Рис. 11. Световая трубка в оптической среде с тензором преломления (4.7).

4.4. Геодезические в гравитационном поле

Описанный авторский пакет программ позволяет легко строить геодезические линии и в 4-мерных псевдоевклидовых римановых пространствах. При этом геодезические линии и их трубки в четырехмерном пространстве по выбору пользователя могут быть представлены в одной из четырех трехмерных проекций четырехмерного пространства. Заметим, что динамическая визуализация именно геодезических трубок позволяет выявить особенности гравитационного взаимодействия, которые могут быть не замечены при исследовании одиночной геодезической линии. Одними из таких особенностей могут являться перекручивание, расширение (сжатие) геодезических и узлы (фокусы), которые могут сопровождать эти процессы. Рассмотрим пример построения и анимации геодезических и их трубок в поле Шварцшильда:

$$\|g_{ik}\| = \begin{pmatrix} 1 - \frac{1}{r_g} & 0 & 0 & 0 \\ 0 & -\frac{1}{1 - \frac{1}{r_g}} & 0 & 0 \\ 0 & 0 & r^2 \sin^2 \theta & 0 \\ 0 & 0 & 0 & -r^2 \end{pmatrix}, \quad (4.8)$$

Зададим метрику (4.8) с помощью имени `gg`

```
gg:=[[-1/(1-2/r),0,0,0],[0,-r^2,0,0],[0,0,-r^2*sin(theta)^2,0],[0,0,0,1-2/r]]:
```

и создадим анимацию геодезической в поле Шварцшильда:

```
A1:=geodesic4d([r,theta,phi,t],s,gg,[R,Theta,Phi,T],
[40,0,0,0],[0,0,0,1],4,5,38,1,0): A1;
```

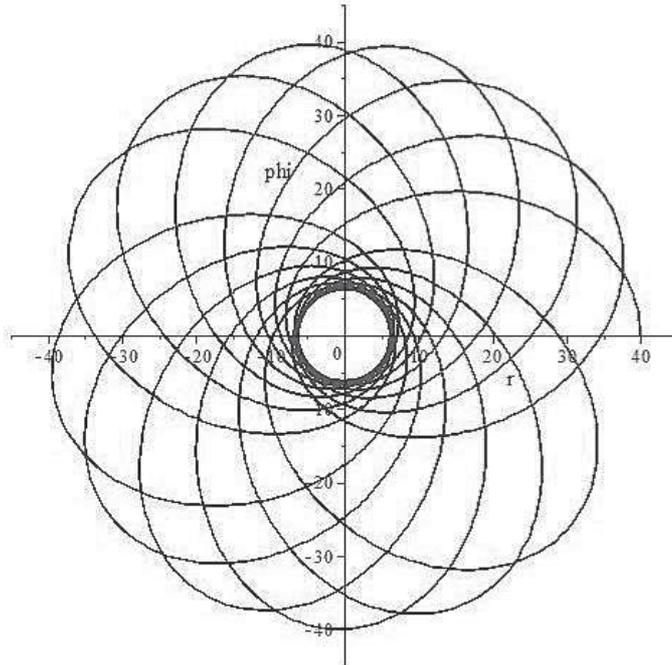


Рис. 12. Геодезическая линия в поле Шварцшильда (4.8), – 150-й кадр. Ясно виден эффект прецессии орбиты.

А теперь создадим анимацию геодезической трубки командой `A11`, полагая, что все орбиты трубки лежат в одной плоскости:

```
A11:=geodesic4d([r,theta,phi,t],s,gg,[R,Theta,Phi,T],
[40,0,0,0],[0,0,0,1],4,5,38,1,1): A11;
```

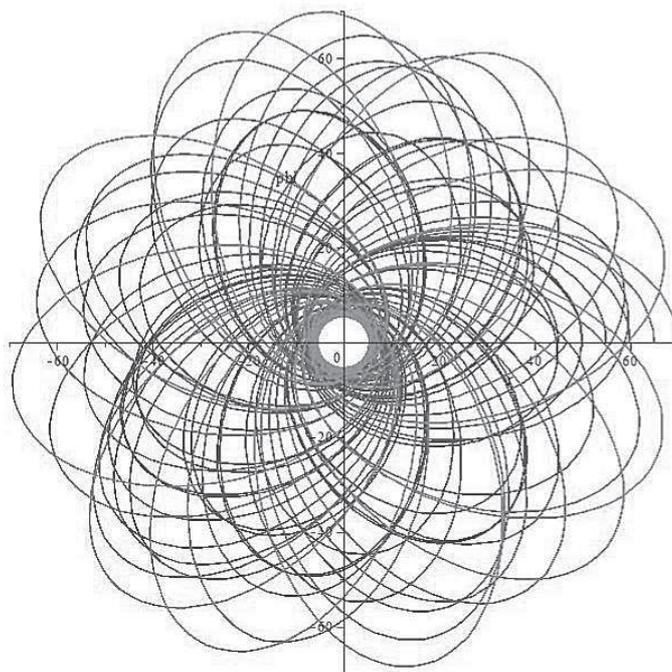


Рис. 13. Геодезическая трубка в поле Шварцшильда (4.8), – 150-й кадр. Ясно видны узлы сгущения и переплетения геодезических.

Заключение

Таким образом, создан достаточно простой и многоцелевой программный комплекс, позволяющий быстро и эффективно осуществлять численное моделирование кривых в 2-, 3- и 4- мерных пространствах. Данный комплекс программ удобно использовать, как в учебных целях, для демонстрации достаточно сложных нелинейных объектов в курсах дифференциальной и римановой геометрии, оптики, теории гравитации, так и для проведения научных исследований в этой области. Комплекс программ свободно распространяется – для получения его с соответствующими инструкциями достаточно обратиться к Авторам. Ссылка при использовании комплекса обязательна.

СПИСОК ЛИТЕРАТУРЫ

1. Самарский А. А., Михайлов А. П. Математическое моделирование: Идеи. Методы. Примеры. – 2-е изд., испр. – М.: Физматлит, 2005. – 320 с.
2. Матросов А.В. Maple 6. Решение задач высшей математики и механики. – СПб.: БХВ-Петербург. – 2001. – 528 с.
3. Дьяконов В.П. Maple 7: Учебный курс. – СПб.: Питер. – 2002. – 672 с.
4. Дьяконов В.П. Maple 9.5/10 в математике, физике и образовании. – М: Солон-пресс. – 2006. – 720 с.
5. Аладьев В.З. Основы программирования в Maple. – Таллинн: Международная Академия Носферы, – 2006. – 301 с.
6. Gray A. Modern Differential Geometry of Curves and Surfaces with Mathematica, Second Edition. – New-York: CRC Press. – 1997. – 432 p.
7. Аладьев В.З., Бойко В.К., Ровба Е.А. Программирование в пакетах Maple и Mathematica: Сравнительный аспект. – Гродно: Изд-во Гродненский госуниверситет, Беларусь, – 2011. – 517 с.
8. Игнат'ев Ю.Г. Проблемы информационных технологий в математическом образовании. – Казань: Изд. - во ТГГПУ, 2005. – 118 с.

9. Кирсанов М.Н. Практика программирования в системе Maple. – М.: Издательский дом МЭИ. – 2011. – 208 с.
10. Игнатъев Ю.Г. Пользовательские графические процедуры для создания анимационных моделей нелинейных физических процессов // Системы компьютерной математики и их приложения. Материалы 10-й международной конференции. – 2009. – вып. 10. – Смоленск: изд-во СмолГУ. – С. 43–44.
11. Игнатъев Ю.Г., Абдулла Х.Х. Математическое моделирование нелинейных обобщенно-механических систем в системе компьютерной математики Maple // Известия ВУЗов, Поволжский регион, физ.-мат. науки. – 2010. – №2 (14) – С. 66–75.
12. Игнатъев Ю.Г., Абдулла Х.Х. Математическое моделирование нелинейных электродинамических систем в системе компьютерной математики Maple // Вестник ТГГПУ. – 2010. – вып. 2(20) – С. 22–27.
13. Игнатъев Ю.Г., Абдулла Х.Х. Математическое моделирование нелинейных обобщенно - механических систем в системе компьютерной математики MAPLE // Вестник РУДН. – 2010. – № 4(3). – С. 65–76.
14. Игнатъев Ю.Г., Исрафилова Э.Г. Математическое моделирование объектов дифференциальной геометрии кривых в системе компьютерной математики MAPLE // Вестник ТГГПУ. – 2011. – № 4(26). – С. 11–16.
15. Игнатъев Ю.Г. Математическое и компьютерное моделирование фундаментальных объектов и явлений в системе компьютерной математики Maple. Лекции для школы по математическому моделированию. // Казань: Казанский университет. 2014. – 298 с. (ISBN 978-5-00019-150-7).
16. Черкасова В.В. Мультипликативный интеграл в дифференциальной геометрии и прикладных задачах. // Вестник ТГГПУ. – 2010. – № 3(21). – С. 57–62.
17. Игнатъев Ю.Г., Черкасова В.В. Программа динамической визуализации процесса качения шара произвольного радиуса вдоль произвольной кривой произвольной параметризованной поверхности с динамическим выводом числовой информации о моментальных координатах векторов адаптированного репера с началом в центре шара, гауссовой и средней кривизн, а также матрицы преобразования репера между начальным и конечным положениями шара. // св. о гос. рег. прог. для ЭВМ Рос. Фед. № 2012615999, 3 июля 2012 г. Федеральная служба по интеллектуальной собственности; RU ОБПБТ №3(80). 20.09.2012. - с. 699.
18. Игнатъев Ю.Г., Самигуллина А.Р. Программное обеспечение теории кривых второго порядка в пакете компьютерной математики // Вестник ТГГПУ. – 2010. – вып. 4(26) – С. 24–29.
19. Игнатъев Ю.Г., Самигуллина А.Р. Программа автоматизированного полного исследования общего уравнения второго порядка на плоскости с выводом результатов исследования в табличном и графическом форматах всех элементов кривых, описываемых общим уравнением, включая формулы их преобразования к каноническому виду, изображения директрис, асимптот, фокусов, исходной и преобразованной системы координат, в математическом пакете Maple. // св. о гос. рег. прог. для ЭВМ Рос. Фед. № 2012611664, 14 февраля 2012 г. Федеральная служба интеллектуальной собственности; RU ОБПБТ №2(79). 20.06.2012.
20. Ю.Г. Игнатъев, Э.Г. Исрафилова. Программа автоматизированного полного исследования произвольной пространственной кривой по ее параметрическим уравнениям с выводом графиков ее дифференциальных параметров и оснащенной динамической визуализации движения адаптированного репера в математическом пакете Maple // Свидетельство о государственной регистрации программ для ЭВМ № 2012616071, 3 июля 2012 г. Федеральная служба по интеллектуальной собственности.
21. Игнатъев Ю.Г., Бушкова В.А. Программа автоматизированного построения геодезических линий на произвольной параметризованной поверхности и их оснащенной динамической визуализации с автоматической оптимизацией графических параметров в системе компьютерной математики Maple: св. о гос. рег. прог. для ЭВМ Рос. Фед. № 2012612380 от 30.05.12.

22. Игнат'ев Ю.Г., Бушкова В.А. Программный комплекс автоматизированного нахождения и построения геодезических линий и геодезических трубок в трехмерных (локально евклидовых) римановых пространствах по заданной метрике и их оснащенной динамической визуализации с автоматической оптимизацией графических параметров в системе компьютерной математики Maple.: св. о гос. рег. прог. для ЭВМ Рос. Фед. № 2012613444 от 28.05.13.
23. Игнат'ев Ю.Г., Бушкова В.А. Программный комплекс автоматизированного нахождения и построения геодезических линий и геодезических трубок в четырехмерных (псевдоевклидовых) римановых пространствах по заданной метрике и их оснащенной динамической визуализации с автоматической оптимизацией графических параметров в системе компьютерной математики Maple.: св. о гос. рег. прог. для ЭВМ Рос. Фед. № 2012613445 от 28.05.13.
24. Эйзенхарт Л.П. Риманова геометрия. М: ИЛ. – 1948. – 316 с.; L.P. Eisenhart. Riemannien Geometry. Princeton: Princeton University. – 1926.
25. Норден А.П. Дифференциальная геометрия. М: ГУПИМП РСФР. – 1948. – 216 с.
26. Игнат'ев Ю.Г. Дифференциальная геометрия кривых и поверхностей в евклидовом пространстве: учебное пособие. IV семестр: курс лекций для студентов математического факультета. Казань: Казанский университет. – 2013. – 204 с.: URL:http://libweb.ksu.ru/ebooks/05_120_000327.pdf.

Поступила в редакцию 14.08.2014

Игнат'ев Юрий Геннадьевич, д. ф.-м. н., профессор, кафедра высшей математики и математического моделирования, Институт математики и механики им. Н.И. Лобачевского, Казанский федеральный университет, 420008, Россия, г. Казань, ул. Кремлевская, 35А.

E-mail: ignatev_yu@rambler.ru

Бушкова Виктория Аркадьевна, аспирант, кафедра высшей математики и математического моделирования, Институт математики и механики им. Н.И. Лобачевского, Казанский федеральный университет, 420008, Россия, г. Казань, ул. Кремлевская, 35А.

E-mail: vbushkova@inbox.ru

Yu. G. Ignat'ev, V. A. Bushkova

Computer simulation and the clothing operated dynamic visualization of curves in differential, Riemannian geometry and Theory of Gravitation

Keywords: The mathematical and computer modelling, the clothing, operated dynamic visualisation, differential and Riemannian geometry, the gravitation theory, applied mathematical packages

PACS: 04.20.Cv, 98.80.Cq, 96.50.S 52.27.Ny

Methods of computer modelling of the basic objects of differential geometry and principles of the clothing dynamic visualisation of the basic objects of differential geometry in applied mathematical package Maple are described. Examples of creation of libraries of program procedures of the equipped operated visual dynamic models of the basic objects of differential, Riemann geometry and Theory of Gravitation are resulted.

REFERENCES

1. Samarsky A.A., Mikhailov A.P. *Mathematics Modelling. Ideas. Methods. Samples.* – Moscow, Fizmatlit, 2005.
2. Matrosov A.V. Maple 6. *Solution of problems of higher mathematics and mechanics.* – Sankt-Peterburg: BXV-Peterburg. – 2001.
3. Diakonov V. P. *Maple 7: Training course.* – SpB.: Piter. – 2002.
4. Diakonov V. P. *Maple 9.5/10 in Mathematics, Physics and Education.* – Moscow: Solon-Press. – 2006.
5. Aladjev V. Z. *Programming bases in Maple.* – Tallin: International Academy of NeoSphere. – 2006.
6. Gray A. *Modern Differential Geometry of Curves and Surfaces with Mathematica*, Second Edition. – New-York:CRC Press. – 1997. – 432 p.
7. Aladjev V. Z., Boiko V. K., Rovba E.A. *Programming in Maple and Mathematica Packages: Comparative aspect.* – Grodno: Grodno State University, Belarus. – 2011.
8. Ignat'ev Yu.G. *IT Problems in mathematical education.* – Kazan: TGHPU, 2005.
9. Kirsanov M. N. *Programming Practice in system Maple.* – Moskow: MEI Publ. – 2011.
10. Ignat'ev Yu.G. Systems of Computer Mathematics and its Applications. *Proc. of 10 Int. Conf.*, Smolensk – 2009. – Vol. 10. p. 43–45.
11. Ignat'ev Yu. G., Abdulla H. H. *Izv. Vuzov, Volga Region*, – 2010. – No 2 (14) – p. 66–75.
12. Ignat'ev Yu. G., Abdulla H. H. *Vestnik TGHPU.* – 2010. – No. 2(20) – p. 22–27.
13. Ignat'ev Yu. G., Abdulla H. H. *Vestnik RUFPP.* – 2010. – No 4(3). – p. 65–76.
14. Ignat'ev Yu. G., Israfilova E. G. *Vestnik TGHPU.* – 2011. –No 4(26). – p. 11–16.
15. Yurii G. Ignatyev. *Mathematical modelling of fundamental objects and the phenomena in system of computer mathematics Maple.* Lectures for school on mathematical modelling. Kazan: Kazan University. 2014. - (ISBN 978-5-00019-150-7).
16. Cherkasova V.V. *Vestnik TGHPU.* – 2010. – No 3(21). – p. 57–62.
17. Ignat'ev Yu. G., Cherkasova V.V. License No 2012615999, 3 Jule 2012; RU OBPBT No 3(80). 20.09.2012. – p. 699.
18. Ignat'ev Yu. G., Samigullina A.R. *Vestnik TGHPU.* – 2010. – No 4(26) – p. 24–29.
19. Ignat'ev Yu. G., Samigullina A.R. License No 2012611664, 14 February 2012; RU OBPBT No2(79). 20.06.2012.
20. Ignat'ev Yu. G., Israfilova E. G. License No 22012616071, 3 Jule 2012; RU OBPBT No 3(80). 20.09.2012. - p. 701.
21. Ignat'ev Yu. G., Bushkova V.A. License No 2012612380, 30.05.12; RU OBPBT.
22. Ignat'ev Yu. G., Bushkova V.A. License No 2012613444, 28.05.13; RU OBPBT.
23. Ignat'ev Yu. G., Bushkova V.A. License No 2012613445, 28.05.13; RU OBPBT.
24. L.P. Eisenhart. *Riemannien Geometry.* Princeton: Princet on University. – 1926.
25. Norden A.P. *Differential Geometry.* Moscow: GUPIMP RSSFR. – 1948.
26. Ignat'ev Yu.G. *Differential Geometry of Curves and Surfaces in Euclidean Space.* IV Semester. Lectures Course for Student of Mathematical Department. Kazan: Kazan University. – 2013.:
URL:http://libweb.ksu.ru/ebooks/05_120_000327.pdf.

Received 14.08.2014

Ignat'ev Yurii Gennadievich, Doctor of Physics and Mathematics, Professor, Lobachevsky Institut of Mathematics and Mechanics , Kazan Phederal University, ul. Kremlyovskaya, 35, Kazan, 420008, Russia.

E-mail: ignatev_yu@rambler.ru

Bushkova Victoria Arkadievna, postgraduate studen, Lobachevsky Institut of Mathematics and Mechanics , Kazan Phederal University, ul. Kremlyovskaya, 35, Kazan, 420008, Russia.

E-mail: vbushkova@inbox.ru