
УДК 51+53+54+ 082

ББК 94

Z 40

Wydawca: Sp. z o.o. «Diamond trading tour»

Druk i oprawa: Sp. z o.o. «Diamond trading tour»

Adres wydawcy i redakcji: Warszawa, ul. Wyszogrodzka, 16
e-mail: info@conferenc.pl

Cena (zł.): bezpłatnie

Zbiór raportów naukowych.

Z 40 Zbiór raportów naukowych. „Wpływ badań naukowych. (28.04.2013 - 30.04.2013) - Bydgoszcz: Wydawca: Sp. z o.o. «Diamond trading tour», 2013. - 100 str.
ISBN: 978-83-63620-99-8 (t.9)

Zbiór raportów naukowych. Wykonane na materiałach Międzynarodowej Naukowo-Praktycznej Konferencji 28.04.2013 - 30.04.2013 roku. Bydgoszcz.
Część 9.

УДК 51+53+54+ 082

ББК 94

Wszelkie prawa zastrzeżone.

Powielanie i kopiowanie materiałów bez zgody autora zakazany.

Wszelkie prawa do materiałów konferencji należą do ich autorów.

Pisownia oryginalna jest zachowana.

Wszelkie prawa do materiałów w formie elektronicznej opublikowanych w zbiorach należą Sp. z o.o. «Diamond trading tour».

Obowiązkowa odniesienia do zbioru.

ISBN: 978-83-63620-99-8 (t.9)

"Diamond trading tour" ©

SPIS /СОДЕРЖАНИЕ

**СЕКЦИЈА 21. ФИЗИКИ I МАТЕМАТИКИ.
(ФИЗИКО-МАТЕМАТИЧЕСКИЕ НАУКИ)**

1. Денисова А.В.....	5
МЕТОД «МАЛЫХ ШЕВЕЛЕНИЙ». РЕШЕНИЕ ЗАДАЧ	
2. Шевцов А.Н. , Жунисбеков С., Туркулбаева З.О.....	10
КОМПЬЮТЕРНАЯ МОДЕЛЬ МЕХАНИЗМА ШАРНИРНОГО ЧЕТЫРЕХЗВЕННИКА	
3. Шевцов А.Н. , Жунисбеков С., Туркулбаева З.О.....	15
РАСЧЕТ КООРДИНАТ ТРЕТЬЕГО ШАРНИРА ЧЕТЫРЕХЗВЕННИКА	
4. Шевцов А.Н.....	22
РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ МОДЕЛИ РЫХЛИТЕЛЯ ГРУНТА	
5. Шевцов А.Н., Асанбаева М.М.....	27
ПОСТРОЕНИЕ ЭЛЕМЕНТАРНЫХ АТТРАКТОРОВ ДЛЯ ГРАНИЦ РЕЗКО ВЫДЕЛЯЮЩИХСЯ ОБЪЕКТОВ	
6. Шевцов А.Н., Асанбаева М.М.....	33
РЕАЛИЗАЦИЯ ПОСТРОЕНИЯ IFS ИЗ ЭЛЕМЕНТАРНЫХ АТТРАКТОРОВ НА DELPHI	
7. Шевцов А.Н., Чанбаева А.И., Абдуразаков Б.	38
ПРИКЛАДНЫЕ ЗАДАЧИ НА MAPLE	
8. Шевцов А.Н., Чанбаева А.И., Абдуразаков Б.	44
АЛГОРИТМЫ РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ НА MAPLE	
9. Шевцов А.Н.....	50
РАСЧЕТ СКОРОСТИ ПОДВИЖНЫХ ШАРНИРОВ ЧЕТЫРЕХЗВЕННИКА	
10. Шевцов А.Н., Айтказина А.М., Абдрахимова А.Н.	53
ИССЛЕДОВАНИЕ УСТОЙЧИВОСТИ ПО ЛЯПУНОВУ НА MAPLE	
11. Шевцов А.Н. , Айтказина А.М., Абдрахимова А.Н.	68
ОБОБЩЕННЫЕ ПОКАЗАТЕЛИ ЛЯПУНОВА	
12. Шевцов А.Н., Жунисбеков С.	73
ОПТИМИЗАЦИЯ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ НА DELPHI	
13. Шевцов А.Н., Жунисбеков С.	79
КОМПЬЮТЕРНОЕ РЕШЕНИЕ И АНАЛИЗ СИСТЕМ ЛИНЕЙНЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ПЕРВОГО ПОРЯДКА	
14. Шевцов А.Н., Кеулимжаева Ж.А.....	89
НЕКОТОРЫЕ ОСОБЕННОСТИ ПОЛИМЕРОВ	

ПОД- СЕКЦІЯ 2. Геометрия.

Денисова А.В.

ученица 11 класса,

Муниципальное бюджетное образовательное учреждение
средняя общеобразовательная школа с углубленным изучением отдельных
предметов №103 г. Нижнего Новгорода

МЕТОД «МАЛЫХ ШЕВЕЛЕНИЙ». РЕШЕНИЕ ЗАДАЧ

При испытании геометрических высказываний на правдоподобие, при поиске контрпримеров математик нередко рассуждает так: “Возьму случай, когда высказывание подтверждается, и «пошевелю» одну точку (или отрезок, или какое-либо другое множество точек). Не могу ли я таким способом получить такой случай, когда высказывание не подтверждается?” [2, с. 4]

Задача 1. Из точки P , расположенной внутри выпуклого многоугольника, опускаются перпендикуляры на его стороны или на их продолжения. Условимся основание такого перпендикуляра называть «приятным», если оно принадлежит стороне, и «неприятным» в противном случае. Верно ли, что всякая внутренняя точка любого многоугольника имеет по крайней мере две приятные проекции?

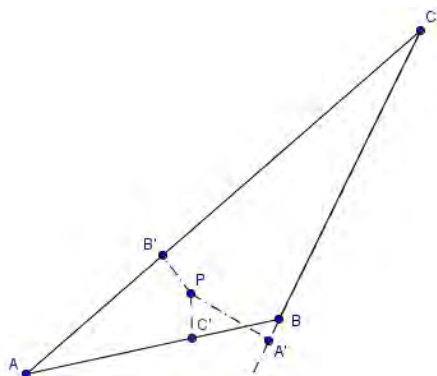


Рисунок 1

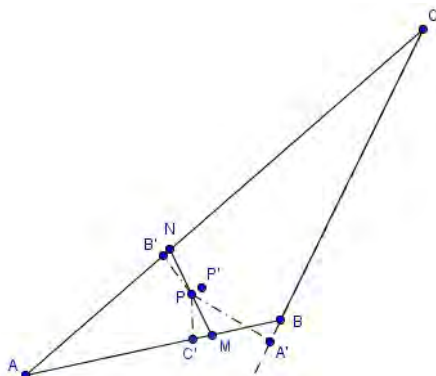


Рисунок 2

Решение:

Возьмем сначала в качестве искомого выпуклого многоугольника треугольник ABC . Если он остроугольный, то все проекции любой его внутренней точки — приятные. Пусть $\triangle ABC$ тупоугольный. В этом случае внутри него легко выбрать такую точку P , у которой будут две приятные проекции (рис. 1). Теперь нетрудно построить выпуклый четырехугольник и точку, лежащую на его контуре, так, чтобы эта точка имела только одну приятную проекцию. Таков, например, четырехугольник $MNCB$ (рис.2 - MN — любая прямая, оставляющая снаружи проекции A' , B' , C'). Точка P на его контуре имеет одну приятную проекцию (этой проекцией будет

сама точка P). Но точка P не лежит внутри многоугольника $MNCB$. Поэтому «пошевелим» точку P — сдвинем ее немного внутрь четырехугольника. При малом «шевелении» точки P мало пошевелятся и ее проекции. Поэтому при достаточно малом шевелении те проекции, которые были неприятными до шевеления точки P , останутся такими и после шевеления. Значит, после шевеления точки P мы получим внутри четырехугольника точку P' , у которой будет ровно одна приятная проекция (а именно — проекция на сторону MN).

Использованный в задаче приём «шевеления» в научной литературе носит название метода «малых шевелений». Приём «малых шевелений» используется для задач, в которых требуется выбрать из некоторого множества фигур ту, которая является наилучшей (берем произвольную фигуру и пытаемся малыми шевелениями ее улучшить, т.е. из нескольких ее свойств изменяем только одно, стараясь сохранить остальные)[2, с.6], а также в задачах, где требуется рассматривать вариации параметров при определенных неизменных условиях. Идею «малых шевелений» полезно привлечь и тогда, когда в каком — то множестве фигур требуется выбрать (построить, найти) ту, которая в том или ином смысле является «наилучшей» (например, найти наименьший периметр, наибольшую площадь и т. п.). На практике этот метод имеет достаточно широкое применение: это решение задач повышенного уровня сложности (в т. ч. олимпиадных) и планиметрические задачи с неоднозначностью в условии, встречающиеся в Едином Государственном Экзамене РФ.

Задача 2 (олимпиада «Росатом» НИЯУ МИФИ, очный тур, 2013г.)

Дана полуокружность радиусом 5 см ($AO=OB=R$). Равнобедренный треугольник EDC ($EC=DE$, DE – основание) располагается внутри нее так, что сторона EC лежит на диаметре, а вершина D принадлежит полуокружности. $EC = 7$ см. Найдите максимально возможную длину DC .

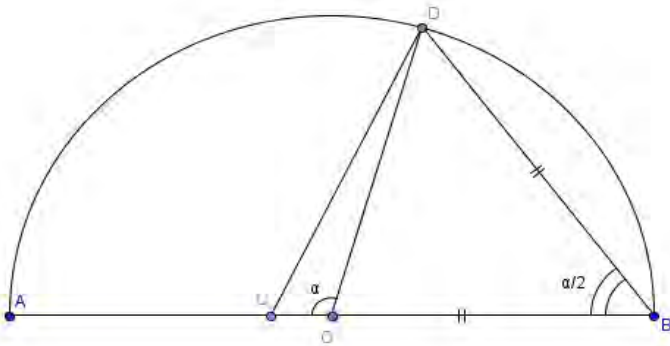


Рисунок 3

Решение:

- 1) Методом ММШ несложно убедиться, что наибольшая длина DC достигается, если точка E совпадает с точкой B (рис.3). Тогда:

$$OC=BC(EC)-OB(R)=2 \text{ (см)}$$

$$DO=R=5 \text{ (см)}$$

- 2) Пусть угол $\angle DOC = \alpha$, а угол DBC соответственно $\alpha/2$ (как вписанный и центральный углы, опирающиеся на одну дугу)

- 3) По теореме косинусов:

$$DC^2=DB^2+BC^2-2DB*BC*\cos(\alpha/2)$$

В то же время:

$$DC^2=CO^2+OD^2-2CO*OD*\cos(\alpha)$$

- 4) Приравнивая правые части уравнений и подставляя численные значения получаем:

$$\cos(\alpha/2)=0,7, \text{ а так же } \cos(\alpha/2)=1,75 \text{ (данное значение невозможно)}$$

- 5) Подставляя полученное значение $\cos(\alpha/2)$ получаем, что DC примерно равно 5,42 ($\sqrt{29,4}$)

Ответ: DC=5,42

Задача 3. В трапеции KLMN известны боковые стороны KL = 36, MN=34 верхнее основание LM =10 и $\cos(\angle KLM) = -1/3$. Найдите диагональ LN.

Решение:

- 1) По условию задачи $\cos(\angle KLM) = -1/3$, значит $\cos(\angle LKN) = -\cos(\angle KLM) = 1/3$.

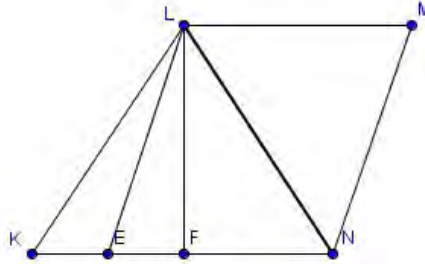
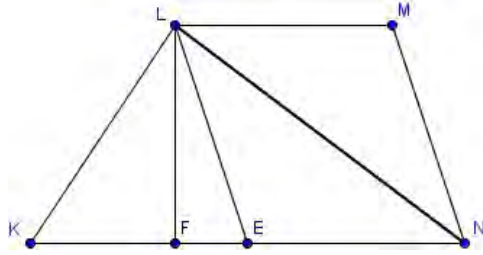
- 2) Из вершины L опустим перпендикуляр к KN (LF).

- 3) Для вычисления LF найдем синус угла LKF.

$$\sin \angle LKF = \sqrt{1 - \frac{1}{9}} = \sqrt{\frac{8}{9}} = \frac{2\sqrt{2}}{3}, \quad LF = 24\sqrt{2}.$$

- 4) Проведем LE \parallel MN, E принадлежит KN.

$$FE = \sqrt{LE^2 - LF^2} = \sqrt{MN^2 - LF^2} = 2.$$



Случай первый (рис.3).

$$FN = FE + EN = 12, LN = 36$$

Случай второй - передвинем точку N (рис.4).

Тогда будем иметь: $FN = EN - FN = 8, LN = 8\sqrt{19}$

Ответ: 36 или $8\sqrt{19}$.

Задача 4. Трапеция ABCD с основаниями AD=6 и BC=4 и диагональю BD=7 вписана в окружность. На окружности взята точка K, отличная от точки D так, что BK=7. Найдите длину отрезка AK (рис.5).

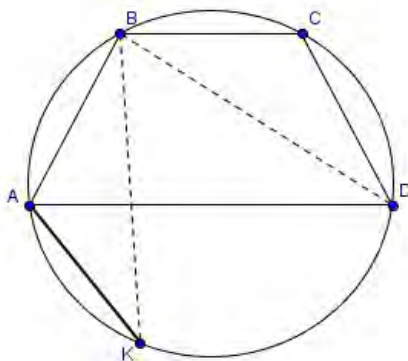


Рисунок 5

Решение:

- 1) Равные отрезки стягивают равные дуги, т.е. дуга BD равна дуге BK .
- 2) Дуга $BD = \text{дуга } BC + \text{дуга } CD$; дуга $BK = \text{дуга } BA + \text{дуга } AK$
- 3) Дуги CD и BA равны (их стягивают стороны p/b

трапеции; трапеция равнобедренная, т.к. только около p/b трапеции можно описать окружность), следовательно дуги AK и BC также равны, т.е. отрезки, стягивающие их имеют одинаковую длину, $AK=BC=4$.

Второй случай невозможен, в чем можно убедиться шевелением точки K по окружности при прочих неизменных параметрах.

Ответ: 4.

Таким образом, метод малых шевелений представляет большой интерес, так как с его помощью можно решать, не прилагая больших усилий, геометрические задачи повышенного уровня сложности. Метод помогает улучшить проективное мышление, а также облегчает возможность нахождения инварианта в задачах с неоднозначностью в условии.

Литература.

1. Канель-Белов А.Я., Ковальджи А.К. - Как решают нестандартные задачи. – М.: МЦНМО, 2001
2. Балк Г., Балк М., Болтянский В. - Метод малых шевелений – М.: МЦНМО, журнал «Квант», 1979, №4

Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»

Таразский Государственный университет им. М.Х.Дулати

Жунисбеков С.

доктор технических наук, профессор, академик НИАРК, ректор

Таразского технического института

Туркумбаева З.О.

студентка 4 курса, специальности 050601-Математика, группа М-29-1,3

Таразский Государственный университет им. М.Х.Дулати

КОМПЬЮТЕРНАЯ МОДЕЛЬ МЕХАНИЗМА ШАРНИРНОГО ЧЕТЫРЕХЗВЕННИКА

Графические методы решения задач механики, при анализе плоских механизмов, наглядны и универсальны, позволяя определять положения, скорости и ускорения звеньев механизмов любой сложности. Но они не могут обладать точностью аналитического решения. А в последнее время требуется еще и компьютерная модель, реализованная в комплексную программу или оболочку, позволяющая получать аналитическое решение, наглядно отображать его на экране и позволять исследовать его при различных начальных условиях.

Рассмотрим механизм шарнирного четырехзвенника [1,с.117], так что оси его звеньев образуют контур ABCD, а положение звеньев определяется углом φ . Аналитическое исследование можно вести методом векторных контуров, разработанным В.А.Зиновьевым (рис.1). Здесь s - переменный по модулю вектор. В соответствии с [1,с.120], нам известны длины звеньев и угол наклона φ , а также закрепленные шарниры – 1 и 4.

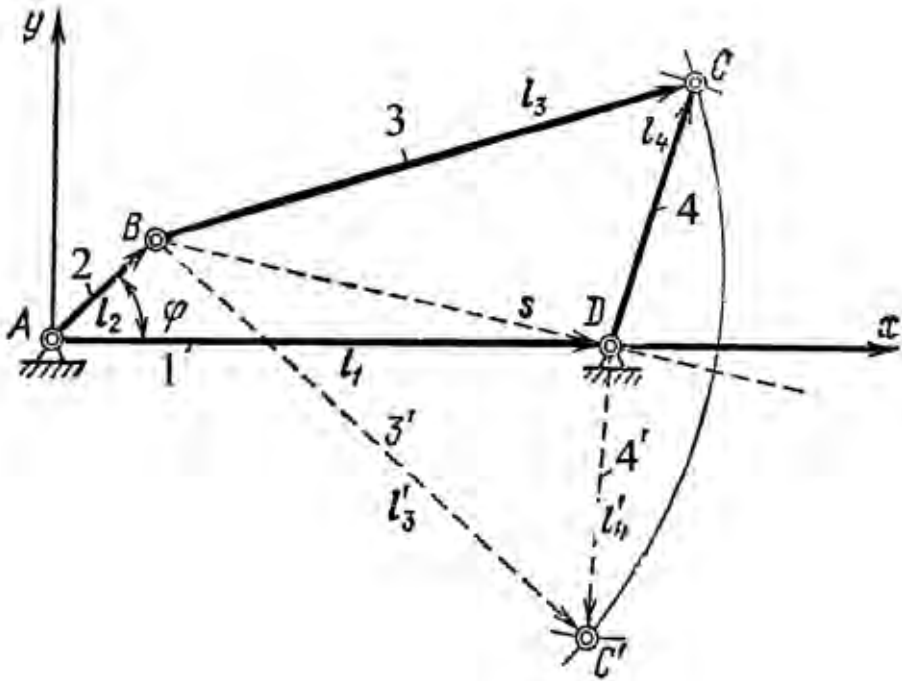


Рисунок 1 – Механизм шарнирного четырехзвенника с двумя векторными контурами.

Разработаем алгоритмы построения для каждого элемента данного механизма:

№	Элемент	Код на Delphi
1	Шарнир	<pre> procedure TForm1.Sharnir(x,y:real); begin series1.AddXY(x,y); series2.AddXY(x,y); end; </pre>
2	Основание	<pre> procedure TForm1.Osnovanie(x,y:real); begin series3.AddXY(x,y-1); series4.AddXY(x,y-3); end; </pre>
3	Звено	<pre> procedure TForm1.zveno(s1,s2:integer); begin series5.AddArrow(L[s1,1],L[s1,2],L[s2,1],L[s2,2]); end; </pre>

Где компоненты TChart, представляют собой приведенные на (рис.2) линии,

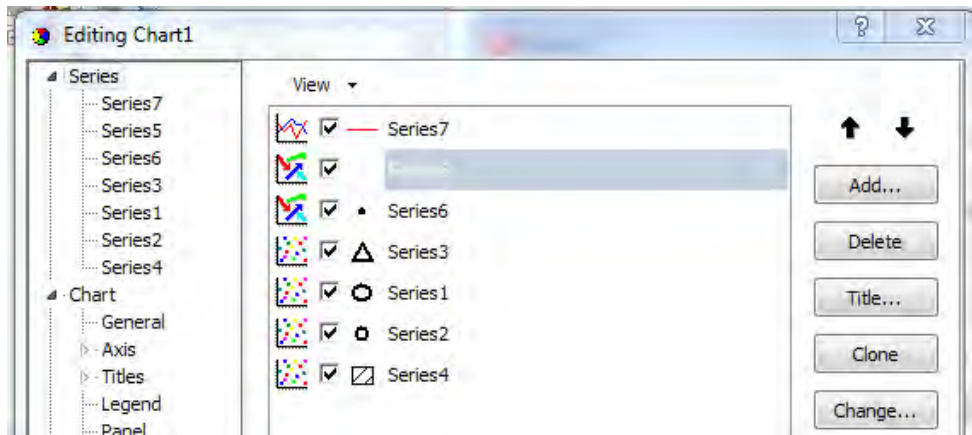


Рисунок 2 – Компоненты TChart.

№	Код на Delphi	Описание
1	begin series1.Clear; series2.Clear; series3.Clear; series4.Clear; series5.Clear;	Очистка данных
2	d[1]:=strtofloat(LabeledEdit1.Text); d[2]:=strtofloat(LabeledEdit2.Text); d[3]:=strtofloat(LabeledEdit3.Text); d[4]:=strtofloat(LabeledEdit4.Text); fi[1]:=strtofloat(LabeledEdit5.Text);	Получение начальных условий
3	L[1,1]:=0; L[1,2]:=0;	Координаты первого шарнира
4	Sharnir(L[1,1],L[1,2]); Osnovanie(L[1,1],L[1,2]);	Отображение первого шарнира
5	L[4,1]:=d[1]; L[4,2]:=0; Sharnir(L[4,1],L[4,2]); Osnovanie(L[4,1],L[4,2]);	Координаты 4-го шарнира и его построение
6	L[2,1]:=d[2]*cos(fi[1]*3.14/180); L[2,2]:=d[2]*sin(fi[1]*3.14/180); Sharnir(L[2,1],L[2,2]);	Нахождение координат 2 шарнира и его построение
7	zveno(1,2);	Построение звена №2
8	S3(L[2,1],L[2,2],d[3],L[4,1],L[4,2],d[4]);	Нахождение координат 3 шарнира
9	zveno(2,3); zveno(3,4); Sharnir(L[3,1],L[3,2]); end;	Построение оставшихся звеньев и шарниров

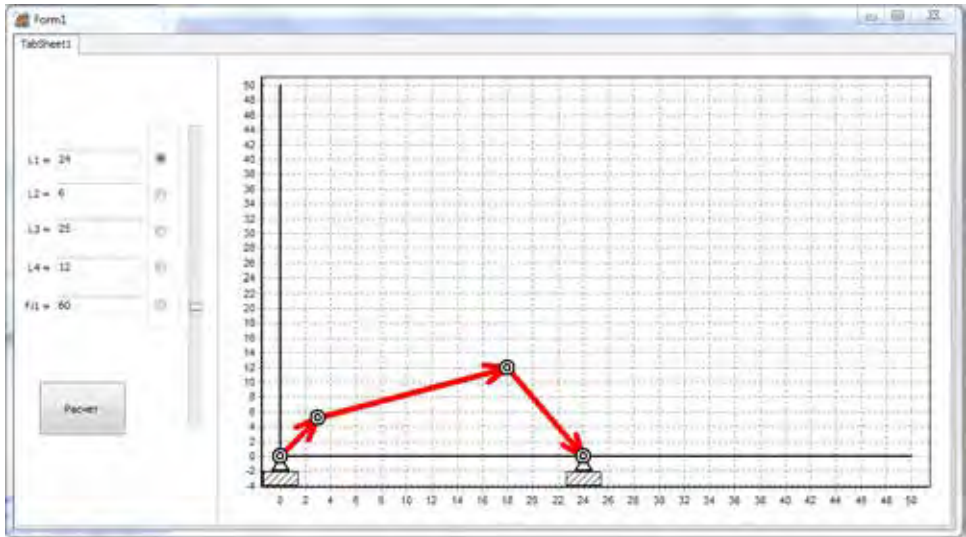


Рисунок 3 – Окно программы.

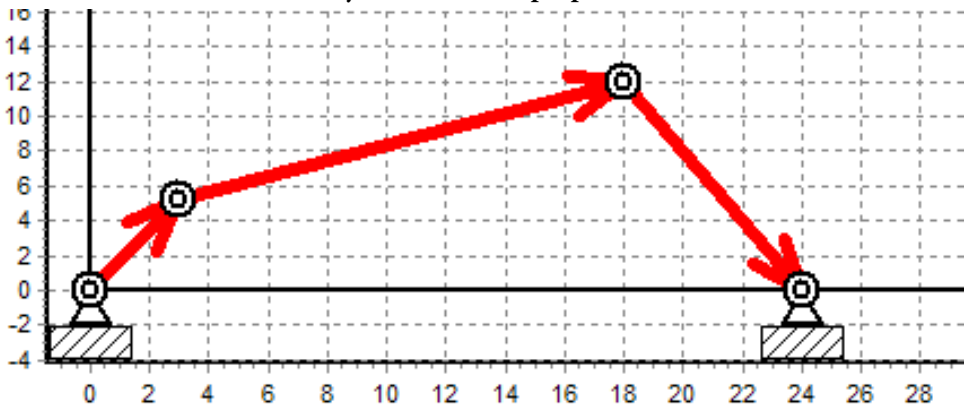


Рисунок 4 – Компьютерная модель четырехзвенника.

Разработанная модель позволяет в автоматическом режиме строить модели различных четырехзвенников находя координаты шарниров (рис.3-4). Для нахождения координат второго шарнира решается система уравнений

$$\begin{cases} x_2 = d_2 \cos(\varphi) \\ y_2 = d_2 \sin(\varphi) \end{cases}$$

третьего шарнира решается система уравнений

$$\begin{cases} (x - x_2)^2 + (y - y_2)^2 = d_3^2 \\ (x - x_4)^2 + (y)^2 = d_4^2 \end{cases}$$

Модель дает представление о структуре четырехзвенника и может быть использована при исследованиях плоских рычажных механизмов.

Литература.

1. Артоболовский И. И. Теория механизмов и машин; Учеб. для вузов. — 4-е изд., перераб. и доп. — М.: Наука. Гл. ред. физ.-мат. лит., 1988. — 640 с.

Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»

Таразский Государственный университет им. М.Х.Дулати

Жунисбеков С.

доктор технических наук, профессор, академик НИАРК, ректор

Таразского технического института

Туркумбаева З.О.

студентка 4 курса, специальности 050601-Математика, группа М-29-1,3

Таразский Государственный университет им. М.Х.Дулати

РАСЧЕТ КООРДИНАТ ТРЕТЬЕГО ШАРНИРА ЧЕТЫРЕХЗВЕННИКА

Математическая модель механизма шарнирного четырехзвенника предполагает закрепление первого и четвертого шарнира [1,с.117]. Два остальных шарнира являются подвижными и степени их подвижности будут зависеть от длин звеньев. Образованный контур ABCD (рис.1), и положение звеньев, определяются углом φ , между осью Oх и вторым звеном. По этим данным довольно просто можно найти положение второго шарнира. Но вот определение координат третьего шарнира представляет довольно сложную задачу приводящую к решению системы:

$$\begin{cases} (x - x_2)^2 + (y - y_2)^2 = d_3^2 \\ (x - x_4)^2 + (y)^2 = d_4^2 \end{cases}$$

Решим задачу на Maple:

```
> restart;
Rd:={(x-x[2])^2+(y-y[2])^2=d[3]^2,
(x-x[4])^2+(y)^2=d[4]^2};
> resh:=solve(Rd,{x,y});
resh[1];
resh[2];
```

Тогда получим следующее решение:

$$Rd := \{(x - x_2)^2 + (y - y_2)^2 = d_3^2, (x - x_4)^2 + y^2 = d_4^2\}$$

$$\begin{aligned}
 x = & -\frac{1}{2} (2 y_2 \text{RootOf}((-8 x_2 x_4 + 4 y_2^2 + 4 x_4^2 + 4 x_2^2) _Z^2 \\
 & + (-4 y_2^3 - 4 d_4^2 y_2 + 8 x_2 x_4 y_2 + 4 d_3^2 y_2 - 4 x_4^2 y_2 - 4 x_2^2 y_2) _Z + y_2^4 + 6 x_2^2 x_4^2 \\
 & + 2 x_2^2 y_2^2 - 4 x_2 x_4 y_2^2 + 4 x_2 x_4 d_3^2 + d_3^4 - 4 x_2^3 x_4 + 4 x_2 x_4 d_4^2 + x_4^4 - 2 x_4^2 d_3^2 + d_4^4 \\
 & + x_2^4 + 2 x_4^2 y_2^2 - 2 x_4^2 d_4^2 - 2 d_4^2 d_3^2 + 2 d_4^2 y_2^2 - 4 x_2 x_4^3 - 2 d_3^2 y_2^2 - 2 x_2^2 d_4^2 \\
 & - 2 x_2^2 d_3^2) + x_4^2 - d_4^2 - x_2^2 - y_2^2 + d_3^2) / (-x_4 + x_2)
 \end{aligned}$$

$$\begin{aligned}
 y = & \text{RootOf}((-8 x_2 x_4 + 4 y_2^2 + 4 x_4^2 + 4 x_2^2) _Z^2 \\
 & + (-4 y_2^3 - 4 d_4^2 y_2 + 8 x_2 x_4 y_2 + 4 d_3^2 y_2 - 4 x_4^2 y_2 - 4 x_2^2 y_2) _Z + y_2^4 + 6 x_2^2 x_4^2 \\
 & + 2 x_2^2 y_2^2 - 4 x_2 x_4 y_2^2 + 4 x_2 x_4 d_3^2 + d_3^4 - 4 x_2^3 x_4 + 4 x_2 x_4 d_4^2 + x_4^4 - 2 x_4^2 d_3^2 + d_4^4 \\
 & + x_2^4 + 2 x_4^2 y_2^2 - 2 x_4^2 d_4^2 - 2 d_4^2 d_3^2 + 2 d_4^2 y_2^2 - 4 x_2 x_4^3 - 2 d_3^2 y_2^2 - 2 x_2^2 d_4^2 \\
 & - 2 x_2^2 d_3^2)
 \end{aligned}$$

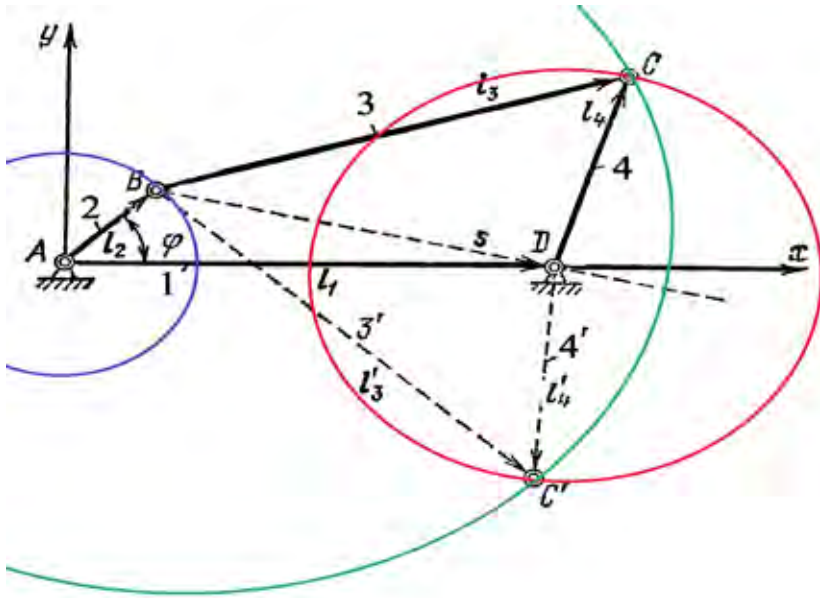


Рисунок 1 – Механизм шарнирного четырехзвенника с двумя векторными контурами.

Преобразуем систему, сложив первое и второе уравнения:

$$\begin{cases} x^2 - 2x_2 + x_2^2 + y^2 - 2y_2 + y_2^2 = d_3^2 \\ x^2 - 2x_4 + x_4^2 + y^2 = d_4^2 \end{cases}$$

$$\begin{cases} 2x(x_4 - x_2) + x_2^2 - x_4^2 - 2y_2 + y_2^2 = d_3^2 - d_4^2 \\ x^2 - 2x_4 + x_4^2 + y^2 = d_4^2 \end{cases}$$

Решаем ее на Maple:

> **restart;**

Rd1:=2*(x[4]-x[2])*x+(x[2])^2-(x[4])^2-2*y[2]*y+(y[2])^2=d[3]^2-d[4]^2;

Rd2:=(x-x[4])^2+(y)^2=d[4]^2;

> **yy:=solve(Rd1,y);**

xx[1]:=solve(Rd2,x)[1];

$xx[2]:=solve(Rd2,x)[2];$

$$Rd1 := 2(x_4 - x_2)x + x_2^2 - x_4^2 - 2y_2y + y_2^2 = d_3^2 - d_4^2$$

$$Rd2 := (x - x_4)^2 + yy^2 = d_4^2$$

$$yy := \frac{1}{2} \frac{d_3^2 - d_4^2 - 2xx_4 + 2xx_2 - x_2^2 + x_4^2 - y_2^2}{y_2}$$

$$xx_1 := \frac{1}{2} \left(-x_2x_4^2 + x_2y_2^2 + d_4^2x_2 - x_4x_2^2 + x_4^3 - d_4^2x_4 + x_2^3 + y_2^2x_4 + d_3^2x_4 - d_3^2x_2 + \right. \\ \left. -4x_2y_2^2d_4^2x_4 - 4x_2y_2^2d_3^2x_4 - 6x_2^2x_4^2y_2^2 + 4x_2x_4^3y_2^2 + 2x_2^2y_2^2d_4^2 + 4x_2^3y_2^2x_4 \right. \\ \left. + 4x_2y_2^4x_4 + 2x_2^2y_2^2d_3^2 + 2d_4^2x_4^2y_2^2 + 2y_2^2x_4^2d_3^2 + 2y_2^2d_3^2d_4^2 - 2x_2^2y_2^4 \right. \\ \left. - x_2^4y_2^2 - x_4^4y_2^2 - 2y_2^4x_4^2 + 2y_2^4d_3^2 + 2y_2^4d_4^2 - y_2^2d_3^4 - y_2^2d_4^4 - y_2^6 \right)^{(1/2)} / \left(\right. \\ \left. -2x_4x_2 + x_4^2 + y_2^2 + x_2^2 \right)$$

$$xx_2 := -\frac{1}{2} \left(x_2x_4^2 - x_2y_2^2 - d_4^2x_2 + x_4x_2^2 - x_4^3 + d_4^2x_4 - x_2^3 - y_2^2x_4 - d_3^2x_4 + d_3^2x_2 + \right. \\ \left. -4x_2y_2^2d_4^2x_4 - 4x_2y_2^2d_3^2x_4 - 6x_2^2x_4^2y_2^2 + 4x_2x_4^3y_2^2 + 2x_2^2y_2^2d_4^2 + 4x_2^3y_2^2x_4 \right. \\ \left. + 4x_2y_2^4x_4 + 2x_2^2y_2^2d_3^2 + 2d_4^2x_4^2y_2^2 + 2y_2^2x_4^2d_3^2 + 2y_2^2d_3^2d_4^2 - 2x_2^2y_2^4 \right. \\ \left. - x_2^4y_2^2 - x_4^4y_2^2 - 2y_2^4x_4^2 + 2y_2^4d_3^2 + 2y_2^4d_4^2 - y_2^2d_3^4 - y_2^2d_4^4 - y_2^6 \right)^{(1/2)} / \left(\right. \\ \left. -2x_4x_2 + x_4^2 + y_2^2 + x_2^2 \right)$$

В соответствии с полученным решением, разработаем алгоритм на Delphi:

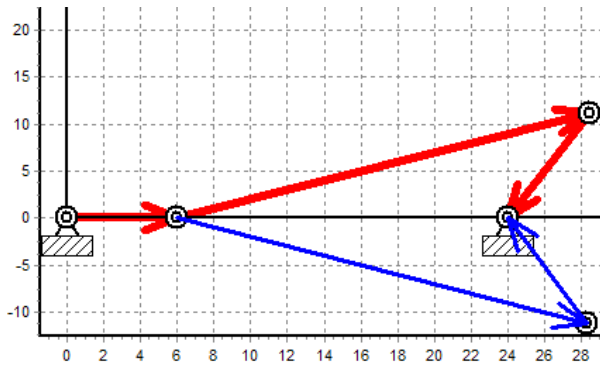
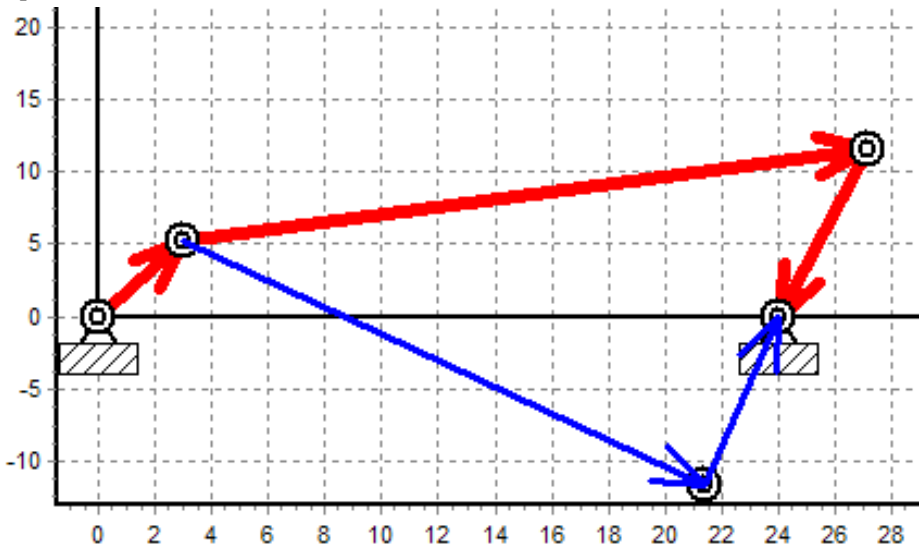
```
procedure TForm1.Mzveno(s1,s2:integer);
begin
series8.AddArrow(L[s1,1],L[s1,2],xx[2],yy[2]);
series8.AddArrow(xx[2],yy[2],L[s2,1],L[s2,2]);
end;

procedure TForm1.S3(x2,y2,d3,x4,y4,d4:real);
var a,b,bb,cc,DD:real;
begin
a:=(x4-x2)/y2;
b:=(sqr(x2)-sqr(x4)+sqr(y2)-sqr(d3)+sqr(d4))/2/y2;
bb:=(a*b-x4)/(sqr(a)+1);
cc:=(sqr(x4)+sqr(b)-sqr(d4))/(sqr(a)+1);
DD:=sqr(bb)-cc;
if DD>=0 then
begin
xx[1]:=-bb+sqrt(DD);      xx[2]:=-bb-sqrt(DD);
yy[1]:=a*xx[1]+b;        yy[2]:=a*xx[2]+b;
L[3,1]:=xx[1];          L[3,2]:=yy[1];
end;
end;

.....
Sharnir(xx[2],yy[2]);
Mzveno(2,4);
.....
```

Wpływ badań naukowych

В результате получим математическую компьютерную модель механизма шарнирного четырехзвенника (рис.2). При изменении параметров звеньев или угла φ , модель автоматически пересчитывает все координаты и перестраивает четырехзвенник.



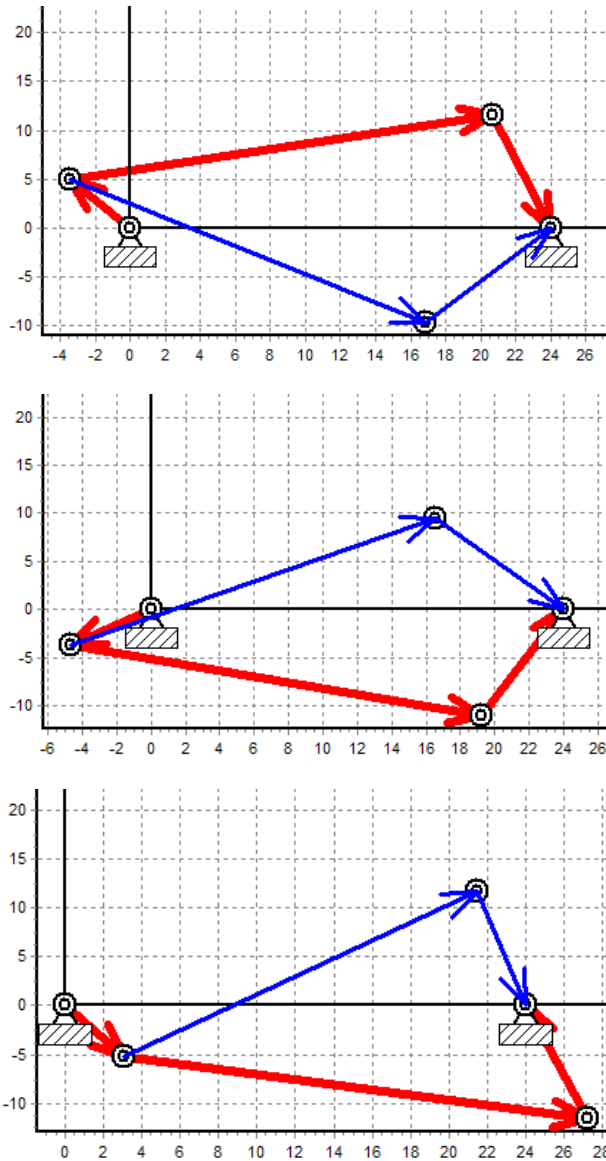


Рисунок 2 – Компьютерная модель механизма шарнирного четырехзвенника

Литература.

1. Артоболовский И. И. Теория механизмов и машин; Учеб. для вузов. — 4-е изд., перераб. и доп. — М.: Наука. Гл. ред. физ.-мат. лит., 1988. — 640 с.

РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ МОДЕЛИ РЫХЛИТЕЛЯ ГРУНТА

В целях изучения рабочего процесса рыхлителя [1], разработаем математическую компьютерную модель для аналитического исследования траектории движения рабочего органа (рис.1).

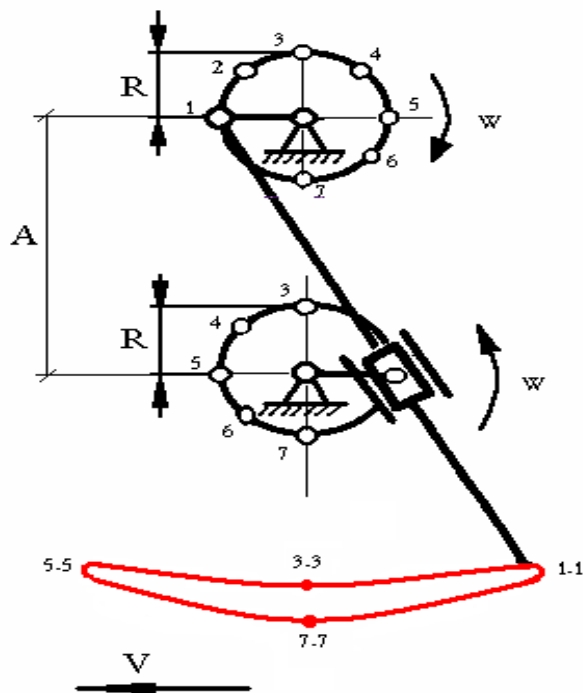


Рисунок 1 – Конструкция рыхлителя.

Математическая модель рыхлителя, задается координатами шестеренок, их радиусами и углом смещения подвижного шарнира на оси второй шестеренки. Красной линией отображается движение рабочего органа рыхлителя при угле смещения 0 градусов от вертикали (положительное направление выбрано по ходу часовой стрелки).

Нам необходимо определять координаты двух шарниров на обоих шестеренках, и по ним уже можно находить точные координаты рабочего органа рыхлителя.

Code: Delphi

```
procedure TForm1.Timer1Timer(Sender: TObject);
begin
timer1.Interval:=10+2*(20-spinedit1.Value) ;
as0:=3.14*(spinedit2.Value)/180 ;
image1.Canvas.FillRect(rect(0,0,1000,1000));
image1.Canvas.CopyRect(rect(0,0,800,800),image2.Canvas,rect(0,0,800,800));
x0:=400; y0:=80;
x1:=400; y1:=200;
koord(x0,y0,x1,y1);
k:=1;
a0:=a0+3.14*(6)/360;
shesterenka(x0,y0,a0,0);
k:=2;
a1:=a1-3.14*(6)/360;
shesterenka(x1,y1,a1,as0);
zzz(zx[1],zy[1],zx[2],zy[2]);
end;
```

Алгоритм каждой шестеренки и ее поворот вокруг оси будем задавать следующим кодом:

Code: Delphi

```
procedure TForm1.shesterenka(x0,y0:integer;a0,as0:real);
begin
R:=50; nz:=15; d:=15;
image1.Canvas.Pen.Color:=clblack; image1.Canvas.Pen.Width:=4;
image1.Canvas.Ellipse(rect(x0-R,y0-R,x0+R,y0+R));
f:= 6.28/nz;
for l := 0 to nz do
begin
fi[1]:= a0+f*i+f/4;
xx[1]:=trunc(x0 +R*sin(fi[1])); yy[1]:=trunc(y0 -R*cos(fi[1]));
fi[2]:= a0+f*(i+1)-f/4;
xx[2]:=trunc(x0 +R*sin(fi[2])); yy[2]:=trunc(y0 -R*cos(fi[2]));
fi[3]:= a0+f*i+f/4+f/8;
xx[3]:=trunc(x0 +(R+d)*sin(fi[3])); yy[3]:=trunc(y0 -(R+d)*cos(fi[3]));
fi[4]:= a0+f*(i+1)-f/4-f/8;
xx[4]:=trunc(x0 +(R+d)*sin(fi[4])); yy[4]:=trunc(y0 -(R+d)*cos(fi[4]));
image1.Canvas.MoveTo(xx[1],yy[1]); image1.Canvas.LineTo(xx[3],yy[3]);
image1.Canvas.LineTo(xx[4],yy[4]); image1.Canvas.LineTo(xx[2],yy[2]);
end;
image1.Canvas.Pen.Width:=2;
image1.Canvas.Ellipse(rect(x0-5,y0-5,x0+5,y0+5));
image1.Canvas.MoveTo(x0-3,y0+3); image1.Canvas.LineTo(x0-10,y0+10);
image1.Canvas.LineTo(x0+10,y0+10); image1.Canvas.LineTo(x0+3,y0+3);
image1.Canvas.Pen.Width:=4; image1.Canvas.Pen.Color:=cllime;
fi[1]:= a0+as0;
zx[k]:=trunc(x0 +(R-3)*sin(fi[1])); zy[k]:=trunc(y0 -(R-3)*cos(fi[1]));
image1.Canvas.Ellipse(rect(zx[k]-5,zy[k]-5,zx[k]+5,zy[k]+5));
end;
```

Рабочий орган, а также связующие звенья зададим следующим образом:

```

procedure TForm1.zzz(x0,y0,x1,y1:integer);
var vx,vy,vpx,vpy:real;
x2,y2,x3,x4,y3,y4,x5,y5,x6,x7,y7,x8,y8,x9,y9:integer;
begin
  image1.Canvas.Pen.Width:=4;
  image1.Canvas.Pen.Color:=clblue;
  image1.Canvas.MoveTo(x0,y0+6);
  vx:=(x1-x0)/sqrt(sqr(x1-x0)+sqr(y1-y0));
  vy:=(y1-y0)/sqrt(sqr(x1-x0)+sqr(y1-y0));
  x2:=x0+trunc(70*vx);
  y2:=y0+trunc(70*vy);
  image1.Canvas.LineTo(x2,y2);
  vpx:=-vy;
  vpy:= vx;
  x3:=x2+trunc(10*vpx);
  y3:=y2+trunc(10*vpy);
  x4:=x2+trunc(-10*vpx);
  y4:=y2+trunc(-10*vpy);
  image1.Canvas.MoveTo(x3,y3);
  image1.Canvas.LineTo(x4,y4);
  x5:=x3+trunc(140*vx);
  y5:=y3+trunc(140*vy);
  x6:=x4+trunc(140*vx);
  y6:=y4+trunc(140*vy);
  image1.Canvas.MoveTo(x3,y3);
  image1.Canvas.LineTo(x5,y5);
  image1.Canvas.MoveTo(x4,y4);
  image1.Canvas.LineTo(x6,y6);
  image1.Canvas.LineTo(x5,y5);
  x7:=trunc((x5+x6)/2);
  y7:=trunc((y5+y6)/2);
  x8:=x7+trunc(120*vx);
  y8:=y7+trunc(120*vy);
  image1.Canvas.MoveTo(x7,y7);
  image1.Canvas.LineTo(x8,y8);
  x9:=x8+trunc(50*(vx+vy));
  y9:=y8+trunc(50*(vy-vx));
  image1.Canvas.LineTo(x9,y9);
  image2.Canvas.Pen.Width:=2;
  image2.Canvas.Pen.Color:=clred;
  if b then image2.Canvas.MoveTo(x9,y9);
  image2.Canvas.LineTo(x9,y9);
  b:=false;
end;

```

Code: Delphi

В результате получим математическую модель рыхлителя (рис.2). Координатную сетку, для определенности, привяжем к расстоянию между центрами шестеренок. Полученные данные и аналитические расчеты дают наглядное представление о траектории рабочего элемента рыхлителя при различных начальных условиях. Разработанная модель также позволяет варьировать все размеры, скорость вращения и угол смещения.

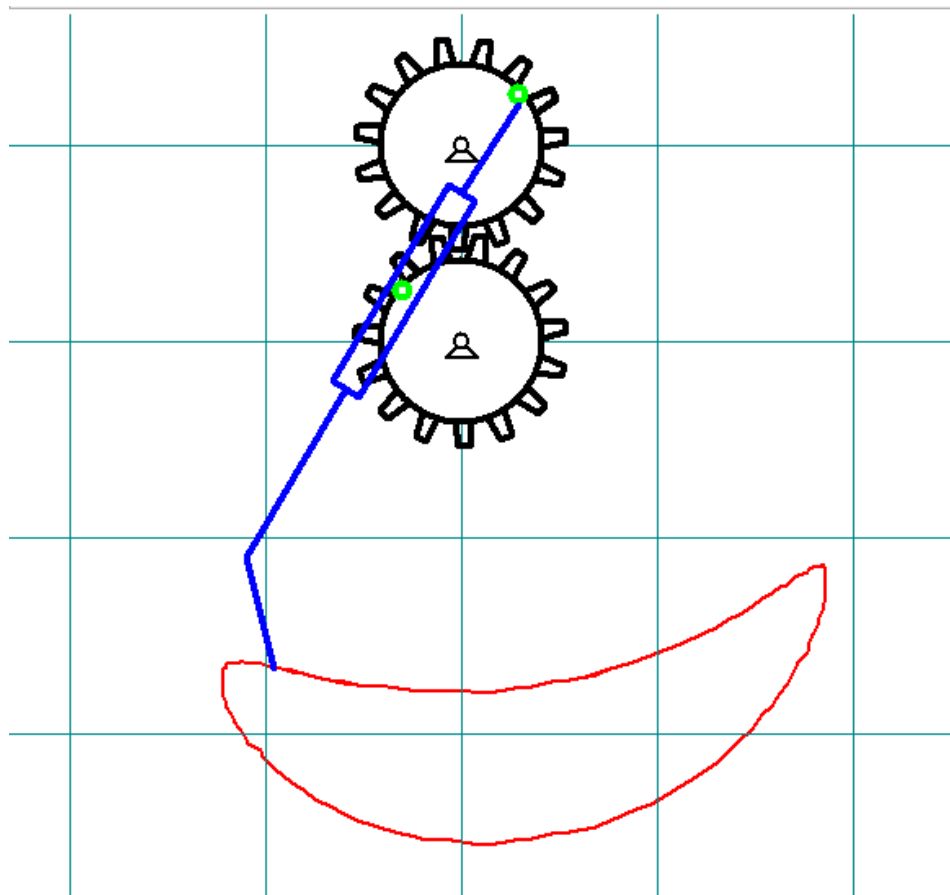


Рисунок 2 – Разработанная математическая модель рыхлителя (угол смещения 0 градусов).

Литература.

1. Антимонов В.П., Бектлеуов А.Ш. Модельные испытания рыхлителя грунта с траекторным колебанием зуба // Водные ресурсы: опыт использования и проблемы. ЖГМСИ, – Тараз, 1997г., С. 176-181.

Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»
Таразский Государственный университет им. М.Х.Дулати

Асанбаева М.М.

студентка 4 курса, специальности 050601-Математика, группа М-29-1,3
Таразский Государственный университет им. М.Х.Дулати

ПОСТРОЕНИЕ ЭЛЕМЕНТАРНЫХ АТТРАКТОРОВ ДЛЯ ГРАНИЦ РЕЗКО ВЫДЕЛЯЮЩИХСЯ ОБЪЕКТОВ

Рассмотрим «Фрактальный алгоритм» применяемый для архивации изображений [1, с.58]. Идея метода основана на том, что мы представляем изображение в более компактной форме — с помощью коэффициентов системы итерируемых функций (Iterated Function System — далее по тексту как IFS). Прежде, чем рассматривать сам процесс архивации, разберем, как IFS строит изображение, т.е. процесс декомпрессии.

Строго говоря, IFS представляет собой набор трехмерных аффинных преобразований, переводящих одно изображение в другое. Преобразованию подвергаются точки в трехмерном пространстве (x координата, y координата, яркость).

Рассмотрим более простой случай – границу некоторого объекта на изображении, получим двумерный случай аффинных преобразований (x координата, y координата). Разработаем программу для выделения границы произвольного объекта, появляющегося на изображении и резко отличающегося от общего фона (рис.1). Теперь очевидно, что найденная граница представляет собой некоторую кривую замкнутую линию толщиной в 1 пиксель (рис.2).

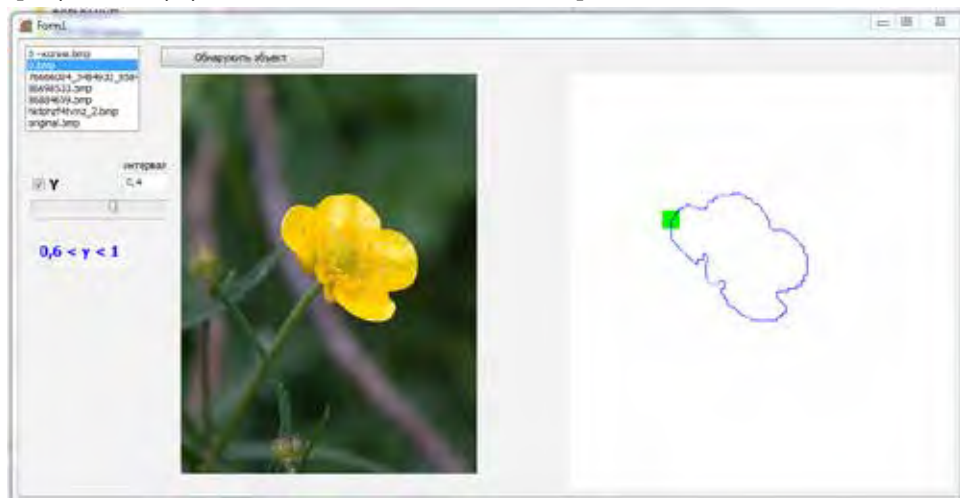


Рисунок 1 – Программа для обнаружения границы резко выделяющегося объекта на изображении.

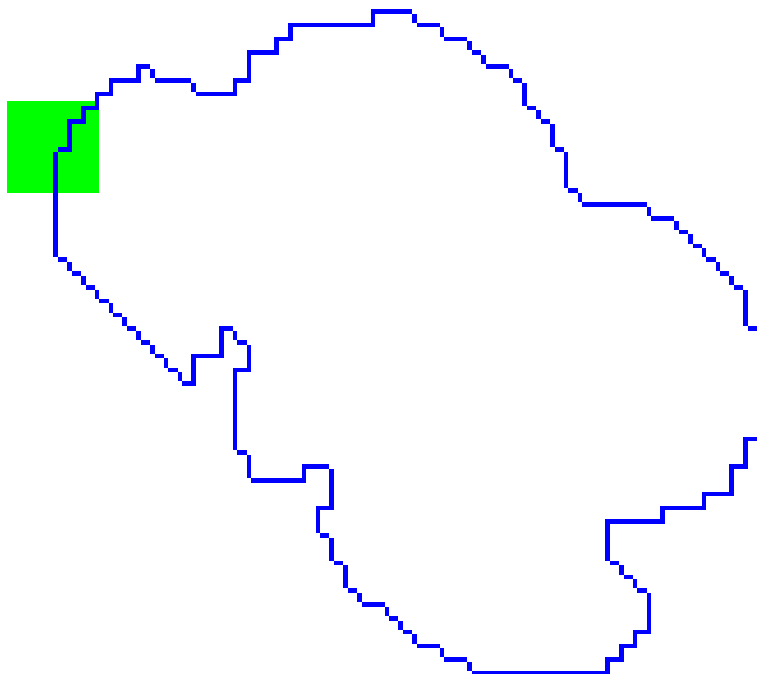


Рисунок 2 – Граница изображения.

Главной задачей для нашей системы IFS станет – поиск аттракторов нашего изображения (координаты x и y , порядок и номер конечной IFS). Необходимо также разработать систему восстановления исходного изображения.

Одна итерация в программе будет заключаться в том, что по исходному изображению с помощью проектирования строится новое, после чего новое берется в качестве исходного [1, с.60]. Утверждается, что в процессе итераций мы получим изображение, которое перестанет изменяться. Оно будет зависеть только от расположения и характеристик линз, и не будет зависеть от исходной картинке. Это изображение называется «неподвижной точкой» или аттрактором данной IFS. Соответствующая теория гарантирует наличие ровно одной неподвижной точки для каждой IFS.

Поскольку отображение является сжимающим, каждая операция в явном

виде задает самоподобные области в нашем изображении. Благодаря самоподобию мы получаем сложную структуру изображения при любом увеличении. Таким образом, интуитивно понятно, что система итерируемых функций задает фрактал (самоподобный математический объект) [2, с.311].

Рассмотрим основные закономерности аффинных преобразований подробно описанные в [3], [4].

Определение. Преобразование $w: R^2 \rightarrow R^2$, представимое в виде

$$w(\lambda) = w \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \bullet \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}$$

где a, b, c, d, e, f - действительные числа и $(x, y) \in R^2$ называется двумерным аффинным преобразованием.

Определение. Пусть $f: X \rightarrow X$ - преобразование в пространстве X . Точка $x_f \in X$ такая, что $f(x_f) = x_f$ называется неподвижной точкой (аттрактором) преобразования.

Определение. Преобразование $f: X \rightarrow X$ в метрическом пространстве (X, d) называется сжимающим, если существует число $s: 0 \leq s < 1$, такое, что $d(f(x), f(y)) \leq s \cdot d(x, y) \quad \forall x, y \in X$.

Теорема (О сжимающем преобразовании).

Пусть $f: X \rightarrow X$ - сжимающее преобразование в полном метрическом пространстве (X, d) . Тогда существует в точности одна неподвижная точка $x_f \in X$ этого преобразования, и для любой точки $x \in X$ последовательность $\{f^n(x): n = 0, 1, 2, 3, \dots\}$ сходится к x_f .

Система аттракторов и их обратных преобразований.








Пусть задано множество некоторых точек, и последовательность обратных сжимающих отображений $f^{-1}: X \rightarrow X$ переводящих аттракторы в конечные множества. Очевидно, что необходимо задать коэффициент сжатия и согласовать сжимающие отображения друг с другом для сохранения замкнутости конечного множества.


Рассмотрим преобразования f^{-1} , для элементарных аттракторов.

Допустим нам дан аттрактор (т.е. две координаты на плоскости (x, y))

преобразование f^{-1} определено коэффициентами (a, b, c, d, e, f) . Занесем все возможные преобразования в таблицу (табл.1).








Таблица 1


№	Аттрактор	Обратное преобразование	
1.		$w_1^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 1 \\ -1 \end{pmatrix}$	$(1, 0, 0, 1, 1, -1)$
2.		$w_2^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix}$	$(1, 0, 0, 1, 1, 0)$
3.		$w_3^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix}$	$(1, 0, 0, 1, 1, 1)$
4.		$w_4^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix}$	$(1, 0, 0, 1, 0, 1)$
5.		$w_5^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} -1 \\ 1 \end{pmatrix}$	$(1, 0, 0, 1, -1, 1)$
6.		$w_6^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \end{pmatrix}$	$(1, 0, 0, 1, -1, 0)$
7.		$w_7^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} -1 \\ -1 \end{pmatrix}$	$(1, 0, 0, 1, -1, -1)$

8.		$w_8^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ -1 \end{pmatrix}$	$(1, 0, 0, 1, 0, -1)$
----	--	---	-----------------------

Существует второе отображение идентичное этому (табл.2).

Таблица 2

№	Аттрактор	Обратное преобразование	
1.		$w_1^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{y} \\ -\frac{1}{x} & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\left(1, \frac{1}{y}, -\frac{1}{x}, 1, 0, 0\right)$
2.		$w_2^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{y} \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\left(1, \frac{1}{y}, 0, 1, 0, 0\right)$
3.		$w_3^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{y} \\ \frac{1}{x} & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\left(1, \frac{1}{y}, \frac{1}{x}, 1, 0, 0\right)$
4.		$w_4^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \frac{1}{x} & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\left(1, 0, \frac{1}{x}, 1, 0, 0\right)$
5.		$w_5^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & -\frac{1}{y} \\ \frac{1}{x} & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\left(1, -\frac{1}{y}, \frac{1}{x}, 1, 0, 0\right)$
6.		$w_6^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & -\frac{1}{y} \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\left(1, -\frac{1}{y}, 0, 1, 0, 0\right)$
7.		$w_7^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & -\frac{1}{y} \\ -\frac{1}{x} & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\left(1, -\frac{1}{y}, -\frac{1}{x}, 1, 0, 0\right)$

8.		$w_8^1 \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -\frac{1}{x} & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\left(1, 0, -\frac{1}{x}, 1, 0, 0 \right)$
----	--	--	--

Выводы. Для границы (рис.2) мы имеем 577 преобразований при одном аттракторе. Полученные преобразования переводят любое замкнутое множество (границу множества), в одну единственную точку (аттрактор), к тому же она – неподвижна. При перемещении аттрактора – перемещается все множество. Принцип построения множества основан на свойствах фракталов, а точнее само подобии, т.к. за основу берется один единственный элемент. Аттрактором может быть не только точка, но и некоторое множество.

Литература.

1.	Книга “Методы сжатия данных”. http://compression.graphicon.ru/
2.	Ватолин Д., Ратушняк А., Смирнов М., Юркин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. – М.: ДИАЛОГ-МИФИ, 2003. – 384с.
3.	Jacquin A. «Fractal image coding based on a theory of iterated contractive image transformations» // Visual Comm. and Image Processing, vol. SPIE-1360, 1990.
4.	Fisher Y. «Fractal image compression» // SigGraph-92.

Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»

Таразский Государственный университет им. М.Х.Дулати

Асанбаева М.М.

студентка 4 курса, специальности 050601-Математика, группа М-29-1,3

Таразский Государственный университет им. М.Х.Дулати

РЕАЛИЗАЦИЯ ПОСТРОЕНИЯ IFS ИЗ ЭЛЕМЕНТАРНЫХ АТТРАКТОРОВ НА DELPHI

Рассмотрим систему итерируемых функций (Iterated Function System — IFS) построенную для элементарных аттракторов. Известно, что для построения границы (рис.1) мы имеем 577 преобразований при одном аттракторе. Принцип построения множества основан на свойствах фракталов, а точнее само подобии, т.к. за основу берется один единственный элемент. Зная все обратные преобразования составим алгоритм на Delphi для их реализации.



Рисунок 1 – Конечное множество.

Для начала просканируем загруженное изображение:

```
procedure TForm1.Scan(bit:Tbitmap);  
var   r1, g1, b1:real; FirstColor, SecondColor:Integer;  
begin   k:=1;  
        for i := 1 to bit.Width do  
        for j := 1 to bit.Height do  
            begin //Получаем цвет текущего пикселя первой картинки  
                FirstColor:=bit.Canvas.Pixels[i,j];  
                //Получаем составляющие RGB  
                r1:=GetRValue(FirstColor);  
                g1:=GetGValue(FirstColor);  
                b1:=GetBValue(FirstColor);  
            if (r1=0)and(g1=0)and(b1=255) then begin  
                at[i,j]:=true; at0.x:=i; at0.y:=j;  
            end; end; end;  
        xmax:=bit.Width;  
        ymax:=bit.Height;  
end;
```

Теперь выделяем границу области и кодируем ее по схеме

7	8	1
6	×	2
5	4	3

одновременно проверяя и заменяя текущую ячейку на пустую.


```

function boool:boolean;
var i,j:integer; b:boolean;
begin
b:=false;
for i := 1 to xmax-1 do   for j := 1 to ymax-1 do
if at[i,j] then b:=true;   boool:=b;
end;
function krest(x,y:integer):integer;
var k:integer;
begin   k:=0;   at[x,y]:=false;
if at[x+1,y] then k:=2;   if at[x,y+1] then k:=4;
if at[x-1,y] then k:=6;   if at[x,y-1] then k:=8;
if k>0 then begin krest:=k;   exit; end;
if at[x+1,y-1] then k:=1;   if at[x+1,y+1] then k:=3;
if at[x-1,y+1] then k:=5;   if at[x-1,y-1] then k:=7;
krest:=k;   end;
procedure TForm1.ScanGr(xy:tpoint);
var i,j,d,kk:integer;
begin   kk:=1;   x00:=xy.x;   y00:=xy.y;
while boool do begin   d:=krest(x00,y00);
if d>0 then   begin   w[kk]:=d;
memo1.Text:=memo1.Text+inttostr(w[kk]);
inc(kk);   nw:=kk;
case d of
1:begin x00:=x00+1;y00:=y00-1; end;   2:begin x00:=x00+1;y00:=y00; end;
3:begin x00:=x00+1;y00:=y00+1; end;   4:begin x00:=x00;y00:=y00+1; end;
5:begin x00:=x00-1;y00:=y00+1; end;   6:begin x00:=x00-1;y00:=y00; end;
7:begin x00:=x00-1;y00:=y00-1; end;   8:begin x00:=x00;y00:=y00-1; end;
end; end;

```

```

end; end;

```


Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»
Таразский Государственный университет им. М.Х.Дулати

Чанбаева А.И.

старший преподаватель кафедры «Прикладная математика»
Таразский Государственный университет им. М.Х.Дулати

Абдуразаков Б.

студент 4 курса специальности «Математика»
Таразский Государственный университет им. М.Х.Дулати

ПРИКЛАДНЫЕ ЗАДАЧИ НА MAPLE

Процесс решения многих прикладных задач механики, физики, экономики и др. очень часто приводит к дифференциальному уравнению или к системе дифференциальных уравнений [1-3]. Именно здесь и целесообразно применение различных систем компьютерной алгебры для быстрого, точного и эффективного их решения. Одной из таких систем является Maple 17. Хотя многие виды дифференциальных уравнений подробно описаны вместе с их решением в [4-5] и др., использование компьютеров значительно упрощает процесс анализа и решения, составленного дифференциального уравнения. Но при всем при этом не может исключить, а лишь дополняет, аналитическое решение.

Рассмотрим некоторые алгоритмы для решения приведенных в [6] прикладных задач.

1) Задача о прямолинейном движении с переменным ускорением [6, с.25] приводится к дифференциальному уравнению первого порядка и решается следующим образом:

> **restart;**

> **dss:=diff(s(t),t\$1)=k/s(t);**

$$dss := \frac{d}{dt} s(t) = \frac{k}{s(t)}$$

> **dss := diff(s(t),t) = k/s(t);**

$$dss := \frac{d}{dt} s(t) = \frac{k}{s(t)}$$

> **ss:=dsolve({dss,s(0)=5},s(t));**

$$ss := s(t) = \sqrt{2 k t + 25}$$

> vv:=diff(ss,t\$1);

$$vv := \frac{d}{dt} s(t) = \frac{k}{\sqrt{2 k t + 25}}$$

> tt:=eval(rhs(vv),t=0)=20;

> kk:=solve(tt,k);

> k:=round(evalf(kk));

> ss;

$$s(t) = \sqrt{200 t + 25}$$

> vv1:=rhs(diff(ss,t\$1));

$$vv1 := \frac{100}{\sqrt{200 t + 25}}$$

> v10:=eval(vv1,t=10);

$$v10 := \frac{4 \sqrt{2025}}{81}$$

> vv10:=evalf(v10);

$$vv10 := 2.222222222$$

> s10:=eval(ss,t=10);

> ss10:=round(evalf(rhs(s10)));

> plot(rhs(vv),t=0..10,thickness=4);

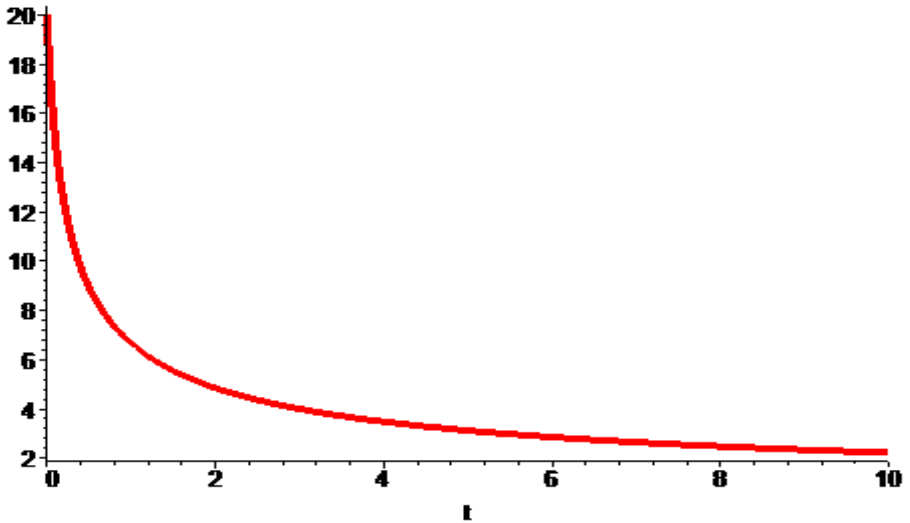


Рисунок 1 - Зависимость скорости тела от времени.

> plot(rhs(ss),t=0..10,thickness=4);

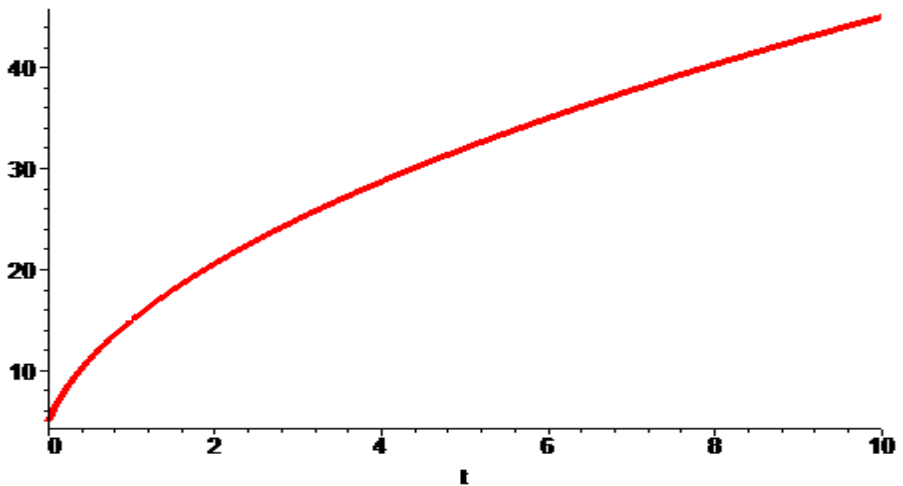


Рисунок 2 - Зависимость пути от времени.

2) Задача криволинейном движении (кривая погони) [6, с.207] приводится к следующему дифференциальному уравнению:

$$\frac{dy}{dx} \cdot 1 + x \frac{d^2y}{dx^2} - \frac{dy}{dx} = -\frac{1}{2} \cdot \frac{dl}{dx},$$

решаем его:

> restart;

> R:=diff(y(x),x\$1)=(y(x)-v*t)/x;

$$R := \frac{d}{dx} y(x) = \frac{y(x) - v t}{x}$$

> R0:=L(x)=2*v*t;

$$R0 := L(x) = 2 v t$$

> R1:=t=solve(R0,t);

$$R1 := t = \frac{1}{2} \frac{L(x)}{v}$$

> R2:=subs(R1,x*R);

$$R2 := x \left(\frac{d}{dx} y(x) \right) = y(x) - \frac{1}{2} L(x)$$

> R3:=diff(L(x),x\$1)=sqrt(1+yy(x)**2);

$$R3 := \frac{d}{dx} L(x) = \sqrt{1 + yy(x)^2}$$

> R4:=expand(diff(R2,x\$1));

$$R4 := \left(\frac{d}{dx} y(x) \right) + x \left(\frac{d^2}{dx^2} y(x) \right) = \left(\frac{d}{dx} y(x) \right) - \frac{1}{2} \left(\frac{d}{dx} L(x) \right)$$

> R5:=simplify(lhs(R4)-rhs(R4)=0);

$$R5 := x \left(\frac{d^2}{dx^2} y(x) \right) + \frac{1}{2} \left(\frac{d}{dx} L(x) \right) = 0$$

> R6:=subs(R3,R5);

$$R6 := x \left(\frac{d^2}{dx^2} y(x) \right) + \frac{1}{2} \sqrt{1 + yy(x)^2} = 0$$

> R7:=diff(y(x),x\$1)=yy(x);

$$R7 := \frac{d}{dx} y(x) = yy(x)$$

> R8:=subs(R7,R6);

$$R8 := x \left(\frac{d}{dx} yy(x) \right) + \frac{1}{2} \sqrt{1 + yy(x)^2} = 0$$

> R9:=dsolve(R8,yy(x));

$$R9 := yy(x) = -\sinh\left(\frac{1}{2} \ln(x) + \frac{-C1}{2}\right)$$

> R10:=dsolve([R8,yy(a)=0],yy(x));

$$R10 := yy(x) = \sinh\left(-\frac{1}{2} \ln(x) + \frac{1}{2} \ln(a)\right)$$

> R11:=lhs(R)=-rhs(R10);

$$R11 := \frac{d}{dx} y(x) = -\sinh\left(-\frac{1}{2} \ln(x) + \frac{1}{2} \ln(a)\right)$$

> R12:=dsolve(R11,y(x));

$$R12 := y(x) = -\frac{(3a-x)\sqrt{x}}{3\sqrt{a}} + _C1$$

> expand(R12);

$$y(x) = -\sqrt{a}\sqrt{x} + \frac{x^{(3/2)}}{3\sqrt{a}} + _C1$$

> R13:=dsolve([R11,y(a)=0],y(x));

$$R13 := y(x) = -\frac{(3a-x)\sqrt{x}}{3\sqrt{a}} + \frac{2a}{3}$$

> R14:=y=subs(x=0,rhs(R13));

> R15:=t=y/v;

> R16:=subs(R14,R15);

$$R14 := y = \frac{2a}{3} \quad R15 := t = \frac{y}{v} \quad R16 := t = \frac{2a}{3v}$$

> with(plots):P[i]:=plot([subs(a=i*10,rhs(R13))], x=0..i*10,thickness=4);

> g:=display(seq(P[i],i=1..20),insequence=true);

display(P[2],P[4],P[6],P[8],P[10], P[12],P[14],P[16],P[18],P[20]);

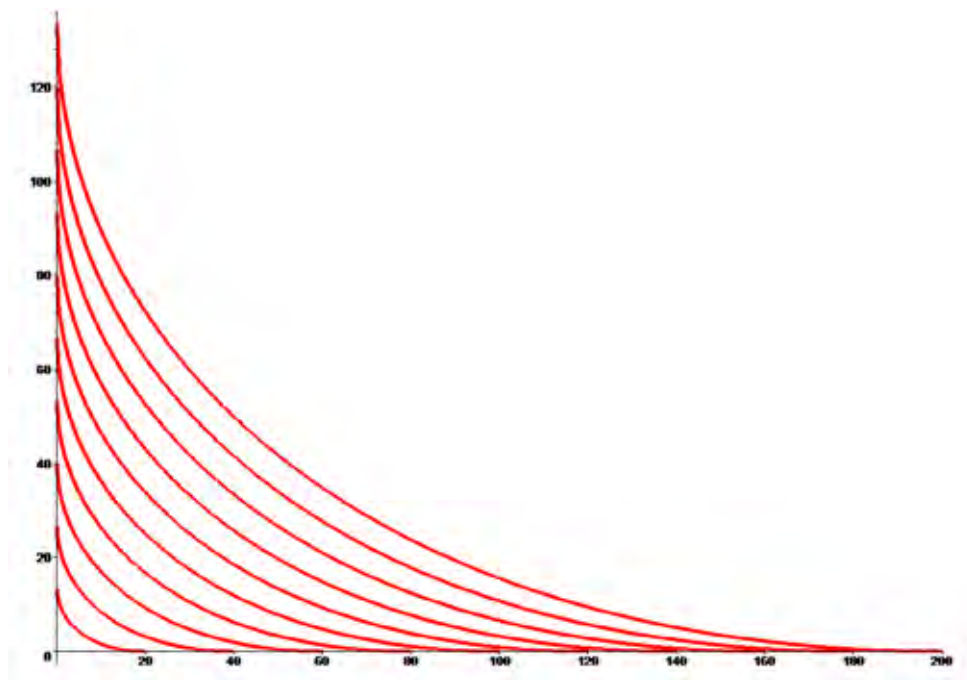


Рисунок 3 – Фазовый портрет полученного решения.

Показанные алгоритмы и программа не являются однотипными и могут меняться в зависимости от условий задачи, полученных начальных условий, составленного дифференциального уравнения и требуемого результата. Подобный метод решения требует знания не только курса дифференциальных уравнений но и навыков работы в Maple. Использование систем компьютерной облегчает поиск решения прикладных задач и способствует лучшему усвоению курса дифференциальных уравнений.

Литература.

1. Пономарев К.К. Составление дифференциальных уравнений. – Минск, «Вышэйшая школа», 1973. -560с.
2. Боголюбов Н.Н., Митропольский Ю.А. Асимптотические методы в теории нелинейных колебаний. – М., 1963. -407 с.
3. Горбузов, В.Н. Целые решения алгебраических дифференциальных уравнений: монография/В.Н.Горбузов.–Гродно:ГрГУ,2006.–255с.
4. Камке Э. Справочник по обыкновенным дифференциальным уравнениям. Пер. с нем. – 4-е изд., испр. — М.: Наука: Гл. ред. Физ-мат. Лит., 1971. — 576с.
5. Зайцев В.Ф., Полянин А.Д. Справочник по обыкновенным дифференциальным уравнениям. – М.: Физматлит, 2001. – 576 с.
6. Шевцов А.Н. Математическое моделирование в прикладных задачах. Алгоритмы программирования Delphi & Maple. – Тараз, 2012. -232 с.

Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»

Таразский Государственный университет им. М.Х.Дулати

Чанбаева А.И.

старший преподаватель кафедры «Прикладная математика»

Таразский Государственный университет им. М.Х.Дулати

Абдуразаков Б.

студент 4 курса специальности «Математика»

Таразский Государственный университет им. М.Х.Дулати

АЛГОРИТМЫ РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ НА MAPLE

Поиск решения дифференциальных уравнений (ДУ) зачастую представляет довольно сложную задачу, а многообразие существующих дифференциальных уравнений [1-2] только усугубляет положение. Для облегчения процессов анализа и решения различных дифференциальных уравнений нами разработана следующая классификационная таблица включающая тип дифференциального уравнения и алгоритм на Maple для его решения.

Рассмотрим основные типы ДУ, и составим алгоритмы для их решения и исследования [3-4].

Таблица 1

№	Тип ДУ	Алгоритм на Maple
1	ДУ первого порядка	<pre>> dsolve(diff(y(x), x)-a*x=0, y(x));</pre> $y(x) = \frac{1}{2} a x^2 + _C1$ <pre>> dsolve(diff(y(x), x)-y(x)=exp(-x), y(x));</pre> $y(x) = \left(-\frac{1}{2} e^{-2x} + _C1 \right) e^x$ <pre>> dsolve(diff(y(x), x)-y(x)=sin(x)*x, y(x));</pre> $y(x) = -\frac{1}{2} \cos(x) x - \frac{1}{2} \cos(x) - \frac{1}{2} \sin(x) x + e^x _C1$

2	ДУ второго порядка	<p>> <code>dsolve(diff(y(x), x\$2)-diff(y(x), x)=sin(x), y(x));</code></p> $y(x) = -\frac{1}{2}\sin(x) + \frac{1}{2}\cos(x) + e^x _C1 + _C2$ <p>> <code>de:=-m*diff(y(x), x\$2)-k*diff(y(x), x);</code></p> $de = m \left(\frac{\partial^2}{\partial x^2} y(x) \right) - k \left(\frac{\partial}{\partial x} y(x) \right)$ <p>> <code>yx0:=y(0)=0, y(1)=1;</code></p> $yx0 = y(0) = 0, y(1) = 1$ <p>> <code>dsolve((de, yx0), y(x));</code></p> $y(x) = -\frac{1}{-1 + e^{\left(\frac{k}{m}\right)x}} + \frac{e^{\left(\frac{k}{m}\right)x}}{-1 + e^{\left(\frac{k}{m}\right)x}}$
3	Решение систем ДУ	<p>> <code>sys1 := [d/dt x(t) = y(t), d/dt y(t) = -x(t)]</code></p> $\text{sys1} := \left[\frac{d}{dt} x(t) = y(t), \frac{d}{dt} y(t) = -x(t) \right]$ <p>> <code>dsolve(sys1)[1]; dsolve(sys1)[2];</code></p> $x(t) = _C1 \sin(t) + _C2 \cos(t)$ $x(t) = _C1 \sin(t) + _C2 \cos(t)$

4	Численное решение	<pre>> dsys := [\frac{d}{dt} x(t) - y(t), \frac{d}{dt} y(t) - x(t), x(0) = 1, y(0) = 0];</pre> <pre>> dsol := dsolve(dsys, numeric) :</pre> <pre>> dsol(1)</pre> <p>$[t = 1., x(t) = 0.540302331778567, y(t) = -0.841471101155307]$</p>
5	ДУ с кусочными функциями	<pre>> eq := diff(y(x), x) + piecewise(x < x^2 - 3, exp(x/2)) * y(x) ;</pre> $eq = \left(\frac{\partial}{\partial x} y(x) \right) + \left\{ \begin{array}{ll} e^{\left(\frac{1}{2}x\right)} & x < x^2 - 3 \\ 0 & \text{otherwise} \end{array} \right\} y(x)$ <pre>> dsolve(eq, y(x)) ;</pre> $y(x) = \begin{cases} -C1 e^{-2e^{\left(\frac{1}{2}x\right)}} & x < \frac{1}{2} - \frac{1}{2}\sqrt{13} \\ -C1 e^{-2e^{\left(\frac{1}{4} - \frac{1}{4}\sqrt{13}\right)}} & x < \frac{1}{2} + \frac{1}{2}\sqrt{13} \\ -C1 e^{-2e^{\left(\frac{1}{4} - \frac{1}{4}\sqrt{13}\right)}} - 2e^{\left(\frac{1}{2}x\right)} + 2e^{\left(\frac{1}{4} + \frac{1}{4}\sqrt{13}\right)} & \frac{1}{2} + \frac{1}{2}\sqrt{13} \leq x \end{cases}$

Кроме функции **dsolve** целесообразно также делать проверку полученного решения:

> **eq:=diff(y(x),x)=piecewise(x>0,x)*y(x)^2;**

$$eq := \frac{\partial}{\partial x} y(x) = \begin{cases} x & 0 < x \\ 0 & \text{otherwise} \end{cases} y(x)^2$$

> **dsolve({y(0)=1,eq},y(x));**

$$y(x) = \begin{pmatrix} \begin{pmatrix} \frac{1}{-CI} & x < 0 \\ -2 \frac{1}{x^2 - 2_CI} & 0 \leq x \end{pmatrix} \end{pmatrix}_{-CI=1}$$

> **simplify(eval(subs(% ,eq))):**

$$\begin{cases} 0 & x \leq 0 \\ 4 \frac{x}{(x^2 - 2)^2} & 0 < x \end{cases} = \begin{cases} 0 & x \leq 0 \\ 4 \frac{x}{(x^2 - 2)^2} & 0 < x \end{cases}$$

Построение фазового портрета позволяет дать больше информации о самом решении чем просто аналитический его вид.

> $DE1 := \left(\frac{d^3}{dx^3} y(x) \right) \cos(x) - \left(\frac{d}{dx} y(x) \right) = y(x)$

> $\text{DEplot}(DE1, y(x), x = 0..1.57, [[y(0) = 1, D(y)(0) = 2, D^{(2)}(y)(0) = 2]])$

$$DE1 := \left(\frac{d^3}{dx^3} y(x) \right) \cos(x) - \left(\frac{d}{dx} y(x) \right) = y(x)$$

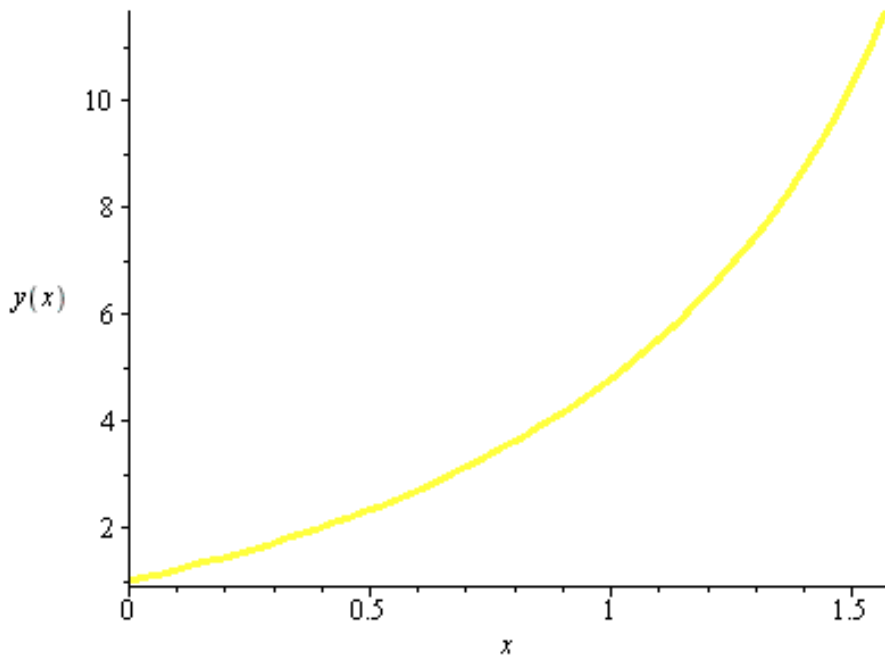


Рисунок 1 – Частное решение ДУ.

>

$$DE3 := \left\{ \frac{d}{dx} y(x) = y(x) - z(x), \frac{d}{dx} z(x) = z(x) - 2y(x) \right\}$$

>

`Display(DE3, [y(x), z(x)], x=0..3, y=0..2, z=-4..4, arrows=large)`

$$DE3 := \left\{ \frac{d}{dx} y(x) = y(x) - z(x), \frac{d}{dx} z(x) = z(x) - 2y(x) \right\}$$

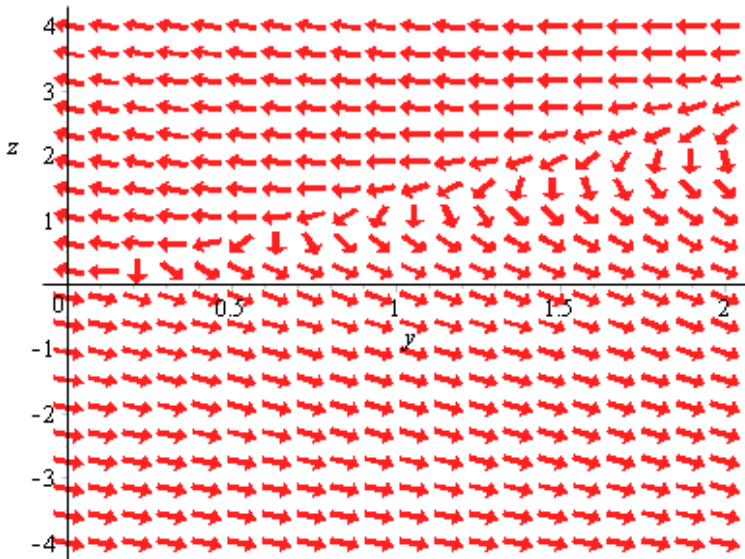


Рисунок 2 – Фазовый портрет ДУ.

Приведенные алгоритмы позволяют достаточно быстро получить решение дифференциального уравнения или системы, сделать проверку решения, построить частное решение и его график, а также определить характер самого решения по фазовому портрету.

Литература.

1. Камке Э. Справочник по обыкновенным дифференциальным уравнениям. Пер. с нем. - М.: Наука: Гл. ред. Физ-мат. Лит., 1971. - 576с.
2. Зайцев В.Ф., Полянин А.Д. Справочник по обыкновенным дифференциальным уравнениям. – М.: Физматлит, 2001. – 576 с.
3. Решение дифференциальных уравнений. -2013. <http://tchernouchkine.narod.ru/maple/tutor/le13.htm>
4. Шевцов А.Н. Математическое моделирование в прикладных задачах. Алгоритмы программирования Delphi & Maple. – Тараз, 2012. -232 с.

РАСЧЕТ СКОРОСТИ ПОДВИЖНЫХ ШАРНИРОВ ЧЕТЫРЕХЗВЕННИКА

Рассмотрим математическая модель механизма шарнирного четырехзвенника [1,с.117]. Два шарнира являются подвижными, а два других закреплены. Зная угловую скорость одного из них будем определять скорость движения второго. Образованный контур ABCD (рис.1), и положение звеньев, определяются углом φ , между осью Ox и вторым звеном. По этим данным построим математическую модель и сделаем расчеты (табл.1)

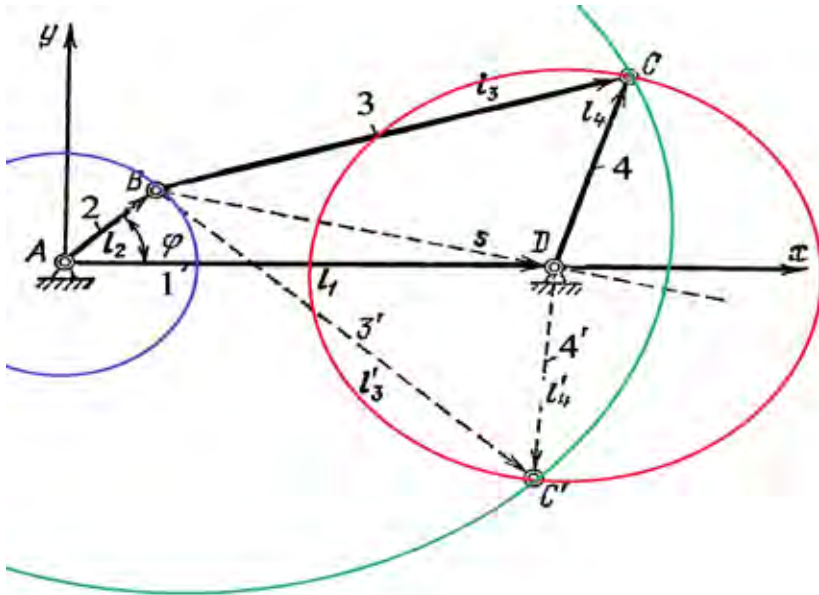


Рисунок 1 – Механизм шарнирного четырехзвенника с двумя векторными контурами.

№	Угол поворота	шарнир С				шарнир С'			
		X1	Y1	путь S1 (м)	V1 (м/с)	X2	Y2	путь S2 (м)	V2 (м/с)
1	350	28,82	-10,98			27,63	11,43		
2	340	29,01	-10,9	0,2061	2,06	26,58	11,71	1,0866	10,86
3	330	28,91	-10,94	0,1077	1,07	25,35	11,92	1,2477	12,47
4	320	28,55	-11,1	0,3939	3,93	24,03	11,99	1,3218	13,21
5	310	27,98	-11,31	0,6074	6,07	22,73	11,93	1,3013	13,01
6	300	27,24	-11,55	0,7779	7,77	21,5	11,73	1,2461	12,46
7	290	26,36	-11,76	0,9047	9,04	20,41	11,45	1,1253	11,25
8	280	25,39	-11,91	0,9815	9,81	19,45	11,1	1,0218	10,21
9	270	24,36	-11,99	1,0331	10,33	18,65	10,74	0,8772	8,77
10	260	23,3	-11,97	1,0601	10,6	17,99	10,38	0,7517	7,51
11	250	22,25	-11,87	1,0547	10,54	17,46	10,06	0,6191	6,19
12	240	21,23	-11,67	1,0394	10,39	17,05	9,78	0,4964	4,96
13	230	20,28	-11,41	0,9849	9,84	16,76	9,57	0,358	3,57
14	220	19,41	-11,08	0,9304	9,3	16,59	9,43	0,2202	2,2
15	210	18,64	-10,74	0,8417	8,41	16,52	9,38	0,086	0,85
16	200	17,99	-10,38	0,743	7,42	16,56	9,41	0,0499	0,49
17	190	17,46	-10,06	0,6191	6,19	16,71	9,53	0,192	1,91
18	180	17,04	-9,78	0,5047	5,04	16,98	9,73	0,336	3,35
20	160	17,88	-10,32	0,602	6,01	16,58	-9,43	0,214	2,13
21	150	18,52	-10,67	0,7294	7,29	16,52	-9,38	0,0781	0,78
22	140	19,26	-11,02	0,8185	8,18	16,56	-9,42	0,0565	0,56
23	130	20,11	-11,35	0,9118	9,11	16,72	-9,54	0,1999	1,99
24	120	21,05	-11,63	0,9808	9,8	16,99	-9,74	0,336	3,35
25	110	22,06	-11,84	1,0316	10,31	17,38	-10	0,4687	4,68
26	100	23,11	-11,96	1,0568	10,56	17,88	-10,32	0,5936	5,93
27	90	24,16	-11,99	1,0504	10,5	18,52	-10,67	0,7294	7,29
28	80	25,2	-11,93	1,0417	10,41	19,3	-11,04	0,8633	8,63
29	70	26,19	-11,79	0,9998	9,99	20,22	-11,39	0,9843	9,84
30	60	27,09	-11,59	0,9219	9,21	21,29	-11,69	1,1112	11,11
31	50	27,86	-11,36	0,8036	8,03	22,5	-11,9	1,228	12,27
32	40	28,46	-11,13	0,6425	6,42	23,79	-11,99	1,2931	12,93
33	30	28,86	-10,96	0,4346	4,34	25,11	-11,94	1,3209	13,2
34	20	29,02	-10,89	0,1746	1,74	26,37	-11,76	1,2727	12,72
35	10	28,88	-10,96	0,1565	1,56	27,46	-11,48	1,1253	11,25
36	0	28,42	-11,15	0,4976	4,97	28,29	-11,2	0,8759	8,75

Получим следующий алгоритм расчета:

begin

radiogroup1.ItemIndex:=4;

trackbar1.Max:=360; inc(tt);

k:=trackbar1.Position-10;

if k<0 then Begin

if checkbox1.Checked then begin timer1.Enabled:=false; exit;end;

k:=k+360; ttmax:=tt; tt:=1;

series9.Clear; series10.Clear;

end;

```
stringgrid1.Cells[1,tt]:=floattostr(k-1);  
trackbar1.Position:=k;  
vr:=abs(timer1.Interval/1000);  
label1.Caption:=floattostr(10/360/(vr))+ ' об./мин.' ;  
timer1.Interval:=spinedit1.Value;  
stringgrid1.RowCount:=ttmax;  
end;
```

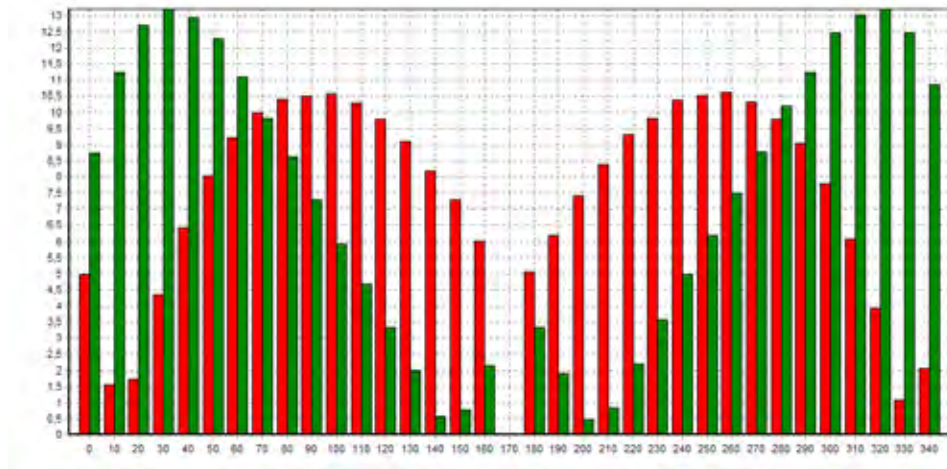


Рисунок 2 – Распределение скоростей подвижных шарниров.

В результате получим следующее распределение скоростей подвижных шарниров (рис.2.). Разработанная модель позволяет меняя начальные параметры получать распределения скоростей для различных угловых скоростей и длин звеньев четырехзвенника.

Литература.

1. Артоболовский И. И. Теория механизмов и машин; Учеб. для вузов. — 4-е изд., перераб. и доп. — М.: Наука. Гл. ред. физ.-мат. лит., 1988. — 640 с.

Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»

Таразский Государственный университет им. М.Х.Дулати

Айтказина А.М.

ст. преподаватель кафедры «Прикладная математика»

Таразский Государственный университет им. М.Х.Дулати

Абдрахимова А.Н.

студентка 4 курса, спец. «Математика»

Таразский Государственный университет им. М.Х.Дулати

ИССЛЕДОВАНИЕ УСТОЙЧИВОСТИ ПО ЛЯПУНОВУ НА MAPLE

Понятие функций Ляпунова появилось в связи с развитием теории устойчивости, начало которой положили труды великого русского математика А.М. Ляпунова. За последние годы наблюдается бурный рост этой теории, вызванный потребностями развивающейся техники, в частности, теории автоматического регулирования и управления.

Развитие теории устойчивости движения осуществляется двумя путями: во-первых, расширением круга задач и, во-вторых, созданием новых и усилением уже известных методов исследования. Метод функций Ляпунова (известный также как второй или прямой метод Ляпунова) является одним из наиболее эффективных методов исследования устойчивости, чем вызвано и его широкое применение в теории управления. Значение его далеко не исчерпывается возможностью установления факта устойчивости или неустойчивости исследуемой системы. Удачно построенная функция Ляпунова для конкретной системы позволяет решать целый комплекс задач, которые имеют важное прикладное значение, например, получение оценки изменения регулируемой величины, оценки времени регулирования, оценки качества регулирования, оценки области притяжения (множества всех начальных возмущений, исчезающих во времени), оценки влияния постоянно действующих возмущений и другие.

Функции Ляпунова позволяют решать вопросы устойчивости в “большом”, т.е. оценивать область начальных возмущений, не выходящих с течением времени за пределы заданной области. С помощью функций Ляпунова решается проблема существования или отсутствия периодических решений, устанавливается ограниченность и продолжимость всех решений заданной нелинейной системы обыкновенных дифференциальных уравнений.

В связи с широким применением функций Ляпунова возник вопрос универсальности этого метода. Решением этой задачи занимались Я.П. Персидский, Н.Н. Красовский, Е.А. Барбашин, Я. Курцвейль, Ж.Л. Массера и другие математики [1-7]. Было установлено, что в теории устойчивости этот метод универсален для широкого круга задач. В этой связи возникла задача о методах построения

функций Ляпунова. Следует заметить, что известные методы построения функций Ляпунова, разработанные для получения достаточных условий устойчивости, не являются достаточно эффективными, поскольку каждый из них приспособлен для исследования конкретных систем. Поэтому проблему построения функций Ляпунова для нелинейных систем в настоящее время нельзя считать решенной.

Данная работа продолжает исследования вопроса о применении функций Ляпунова к исследованию продолжимости решений дифференциальных уравнений.

В данной работе мы будем рассматривать системы дифференциальных уравнений в нормальной форме.

$$\dot{x}^i = X^i(t, x^1, x^2, \dots, x^n) \quad i = 1 \dots, n, \quad (\text{ф.1})$$

Напомним, что система обыкновенных дифференциальных уравнений называется нормальной. В этой системе t --- независимая переменная, x^1, x^2, \dots, x^n --- неизвестные функции этой переменной, а X^1, X^2, \dots, X^n --- функции от $n+1$ переменной, заданные на множестве Γ пространства размерности $n+1$, в котором координатами точки являются числа t, x^1, x^2, \dots, x^n . В дальнейшем будем предполагать, что функции

$$X^i(t, x^1, x^2, \dots, x^n) \quad i = 1 \dots, n, \quad (\text{ф.2})$$

непрерывны на открытом множестве Γ ; также будем предполагать, что их частные производные

$$\frac{\partial X^i(t, x^1, x^2, \dots, x^n)}{\partial x^j}, \quad i, j = 1 \dots, n, \quad (\text{ф.3})$$

существуют и непрерывны на множестве Γ . Следует заметить, что частные производные, непрерывность которых предполагается, берутся только по переменным x^1, x^2, \dots, x^n , а не по независимой переменной t .

Решением системы уравнений (ф.1) называется система непрерывных функций определенных на некотором интервале $t_1 < t < t_2$ и удовлетворяющих системе (ф.1).

$$x^i = \varphi(t) \quad i = 1 \dots, n, \quad (\text{ф.4})$$

Интервал $r_1 < t < r_2$ называется интервалом определения решения (случаи $r_1 = -\infty, r_2 = +\infty$ не исключаются). Считается, что система функций удовлетворяет системе уравнений, если при подстановке в соотношение вместо x^1, x^2, \dots, x^n функций соотношения превращаются в тождества по t на всем интервале $r_1 < t < r_2$ и чтобы правые части уравнений были определены для всех подставляемых в них значений аргументов. Таким образом, точка с координатами $t, \varphi^1(t) \dots, \varphi^n(t)$ должна принадлежать множеству Γ для всех значений t на интервале $r_1 < t < r_2$.

Устойчивость по Ляпунову

Рассмотрим систему дифференциальных уравнений

$$\dot{y} = Y(y, t) \quad (\text{ф.5})$$

Выделим некоторое решение $y = f(t)$ системы (ф.5) и назовем его невозмущенным решением.

Решение $y = f(t)$ назовем устойчивым в смысле Ляпунова, если для любого $\varepsilon > 0$ можно указать $\delta > 0$ такое, что из неравенства $\|y(t_0) - f(t_0)\| < \delta$ следует неравенство $\|y(t) - f(t)\| < \varepsilon$ при $t \geq t_0$. Здесь через $y(t)$ обозначено любое другое решение системы (ф.5), определяемое начальным условием $y(t_0)$. Решение $y = f(t)$ называется асимптотически устойчивым в смысле Ляпунова, если оно устойчиво в смысле Ляпунова и если существует такое $h > 0$, что при $\|y(t_0) - f(t_0)\| < h$ будем иметь

$$\lim_{t \rightarrow \infty} \|y(t) - f(t)\| = 0 \quad (\text{ф.6})$$

Рассмотрим пример:

Решение $x = 0$ уравнения $\dot{x} = x^2$ не является устойчивым ни справа, ни слева, т.к. каждое решение $x = x_0(1 - tx_0)^{-1}$, где $x_0 > 0$ ($x_0 < 0$), перестает существовать при $t = t_0^{-1}$ (рис. 1).

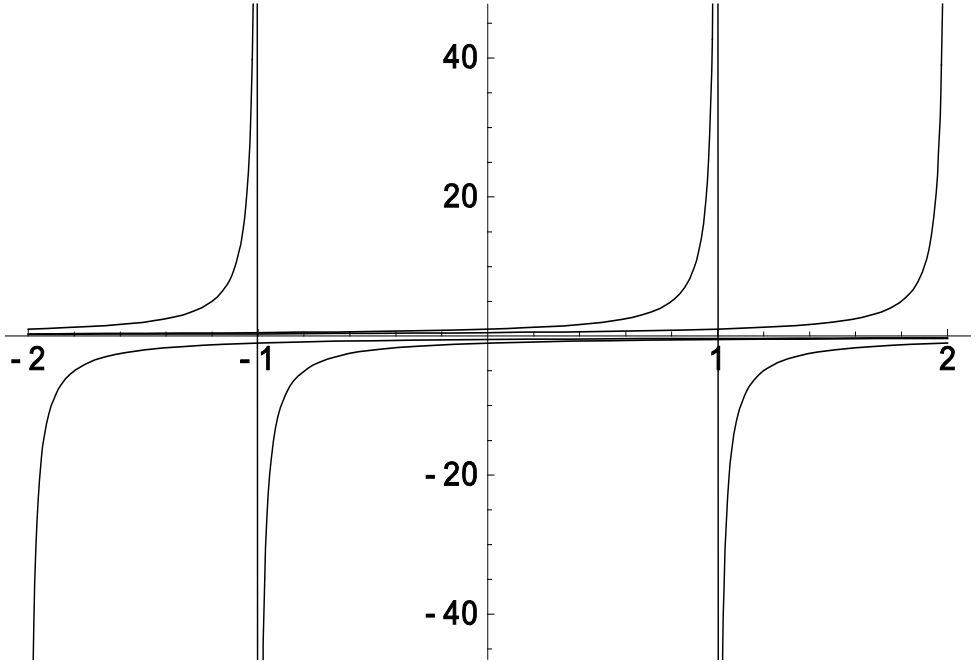


Рис. 1

Рисунок 1 – Неустойчивое решение при $x=0$.

Разработаем программу реализующую поиск решения системы дифференциальных уравнений и построение фазового портрета.

```
> restart;  
dy0:=diff(y(x),x$1)=x^2;
```

$$dy0 := \frac{d}{dx} y(x) = x^2$$

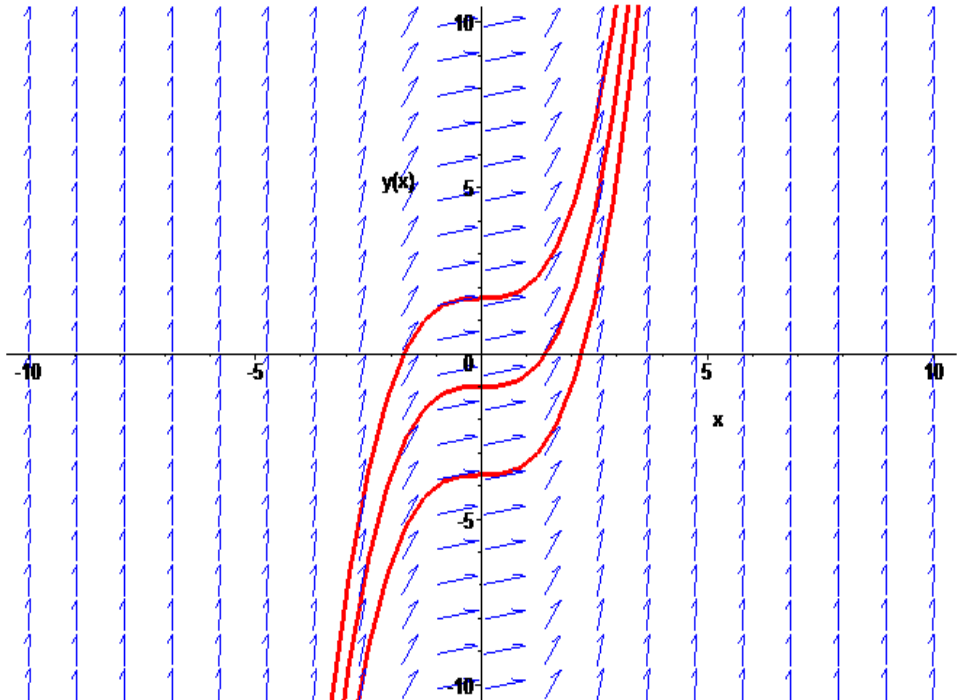
```
> dsolve(dy0,y(x));
```

$$y(x) = \frac{x^3}{3} + C1$$

> with(DEtools):

a:=1;b:=3; DEplot(dy0,y(x),x=-10..10,y=-10..10, [y(0)=-1,y(2)=-1,y(-2)=-1],
colour=blue,linecolor=[red,red,red]);

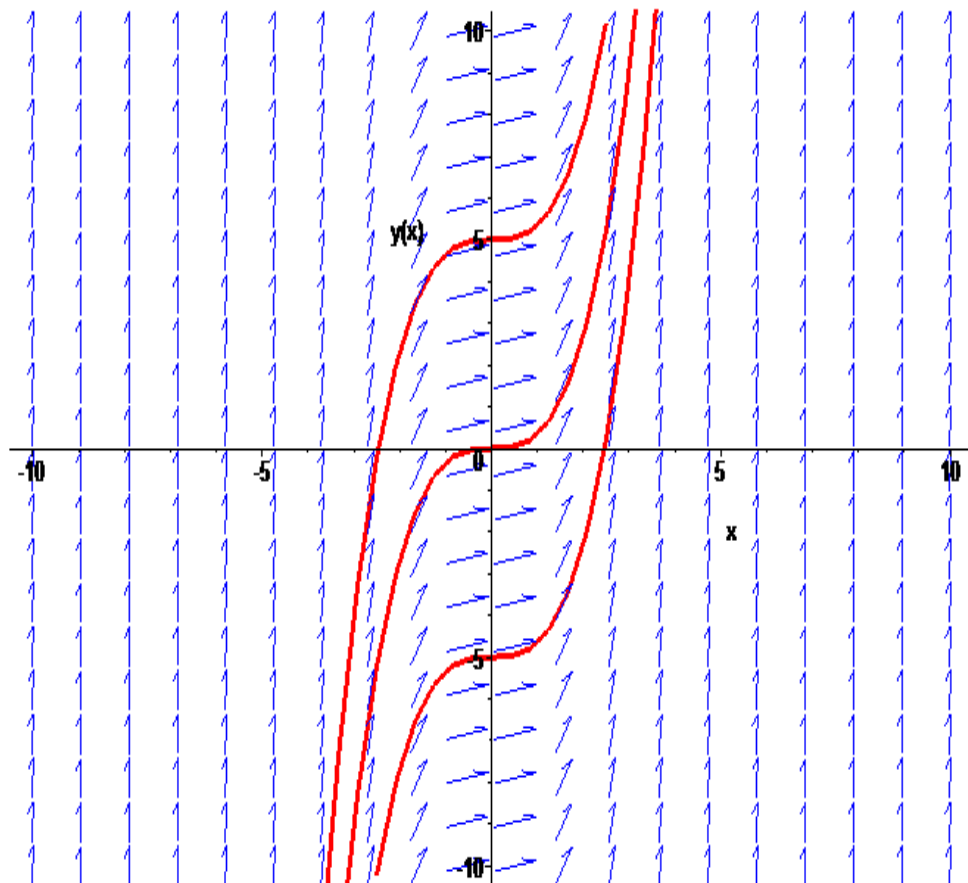
a := 1 b := 3



> with(DEtools):

a:=1;b:=3; DEplot(dy0,y(x),x=-10..10,y=-10..10, [y(0)=5,y(0)=0,y(0)=-5],
colour=blue,linecolor=[red,red,red]);

a := 1 b := 3



> restart;

dy0:=diff(y(x),x\$1)=1-x^2;

$$y(x) = -\frac{1}{3}x^3 + x + _CI$$

> dsolve(dy0,y(x));

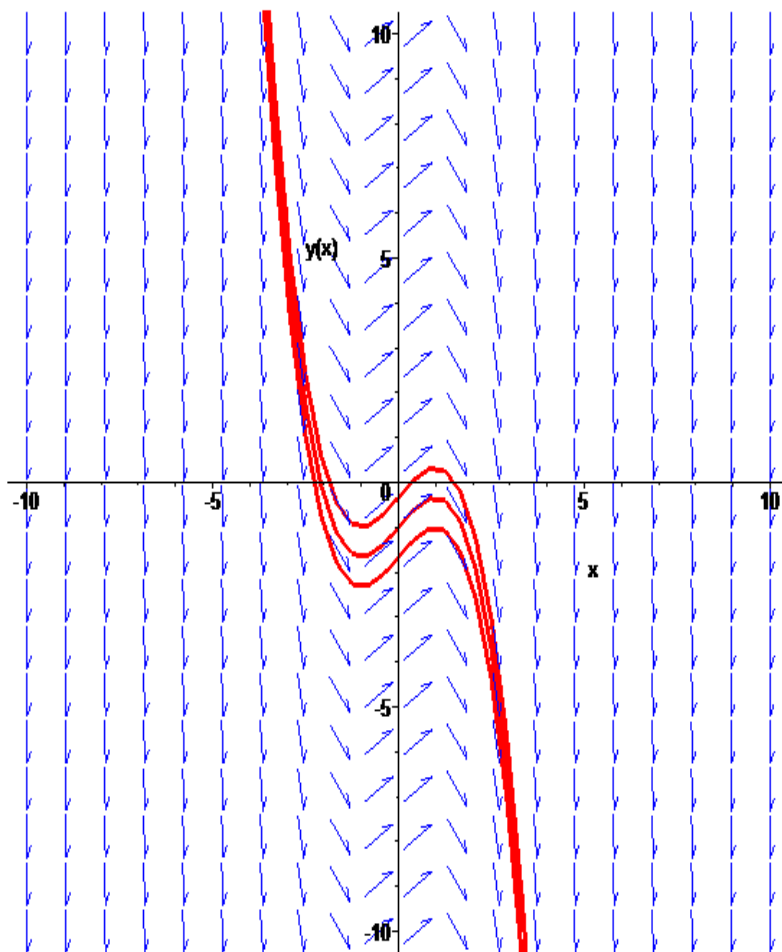
$$y(x) = -\frac{1}{3}x^3 + x + _CI$$

> with(DEtools):

a:=1;b:=3; DEplot(dy0,y(x),x=-10..10,y=-10..10, [y(0)=-1,y(2)=-1,y(-2)=-1],

colour=blue,linecolor=[red,red,red]);

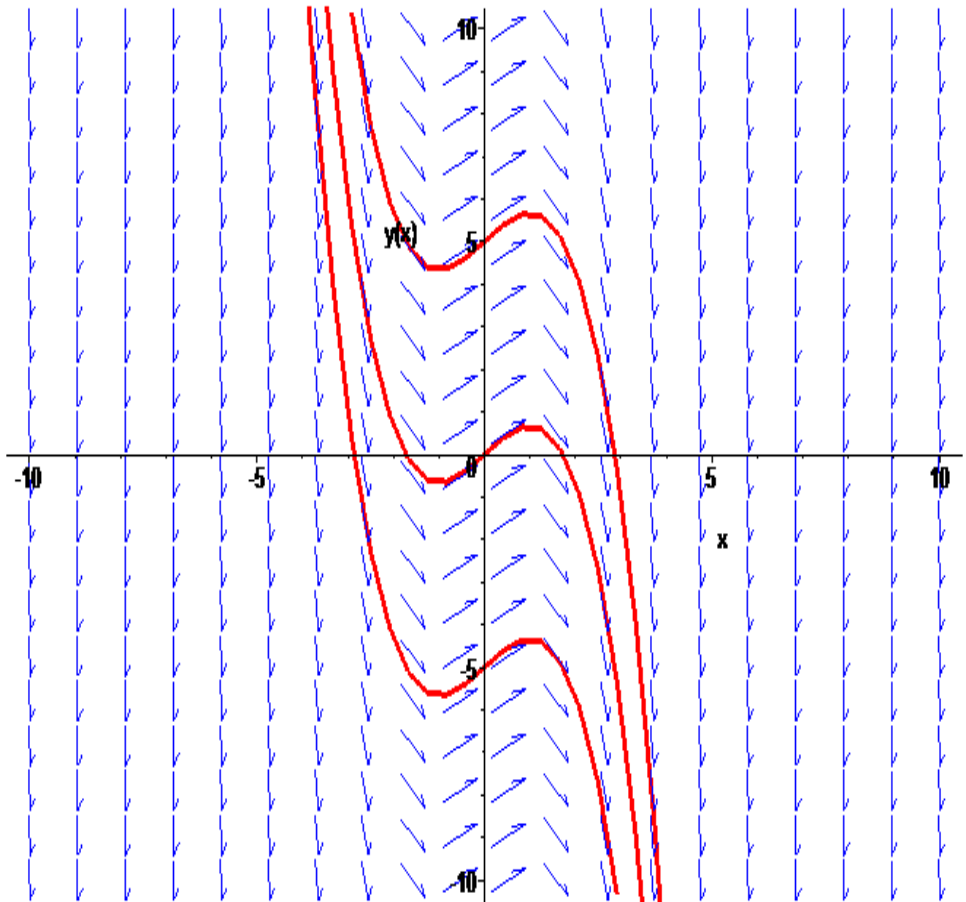
a := 1 b := 3



> with(DEtools):

```
a:=1;b:=3; DEplot(dy0,y(x),x=-10..10,y=-10..10, [y(0)=5,y(0)=0,y(0)=-5],
colour=blue,linecolor=[red,red,red]);
```

```
a := 1    b := 3
```



> restart;

dy0:=diff(y(x),x\$1)=1-x^3;

$$dy0 := \frac{d}{dx} y(x) = 1 - x^3$$

> dsolve(dy0,y(x));

$$y(x) = -\frac{1}{4}x^4 + x + _C1$$

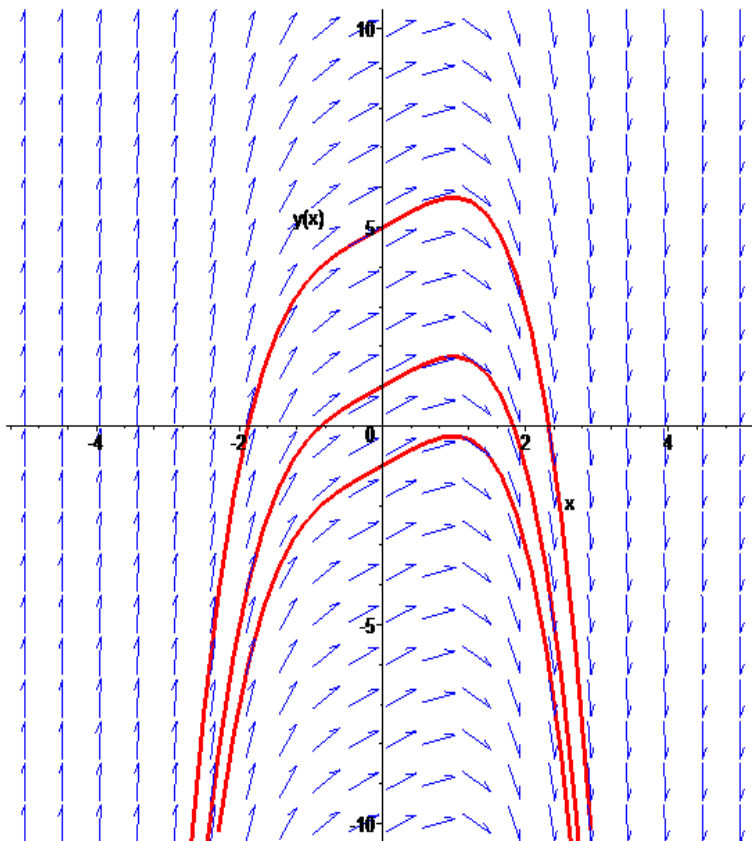
> with(DEtools):

a:=1;b:=3; DEplot(dy0,y(x),x=-5..5,y=-10..10,

[y(0)=-1,y(2)=-1,y(-2)=-1],

colour=blue,linecolor=[red,red,red]);

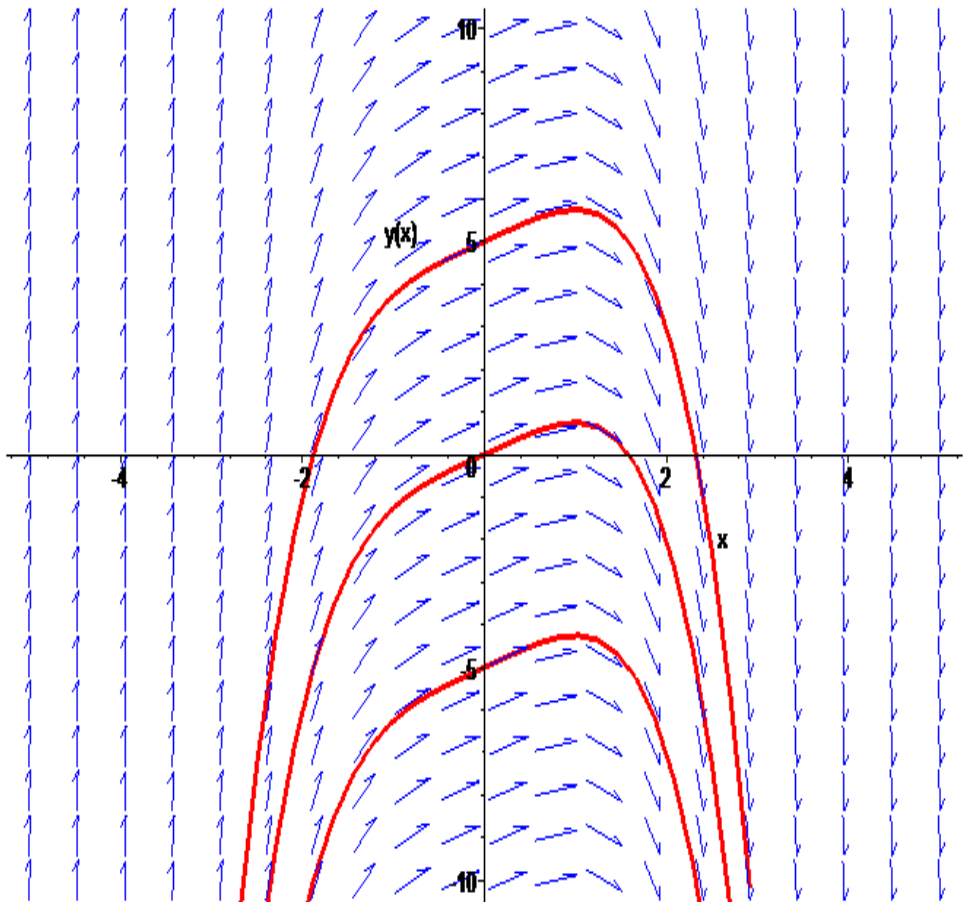
$$b := 3 \quad b := 3$$



> with(DEtools):

```
a:=1;b:=3; DEplot(dy0,y(x),x=-5.5,y=-10..10, [y(0)=5,y(0)=0,y(0)=-5],
colour=blue,linecolor=[red,red,red]);
```

$$b := 3 \quad b := 3$$



> restart;

dy0:=diff(y(x),x\$1)=1-x^2+3*x;

$$y(x) = -\frac{1}{3}x^3 + \frac{3}{2}x^2 + x + _C1$$

> dsolve(dy0,y(x));

$$y(x) = -\frac{1}{3}x^3 + \frac{3}{2}x^2 + x + _C1$$

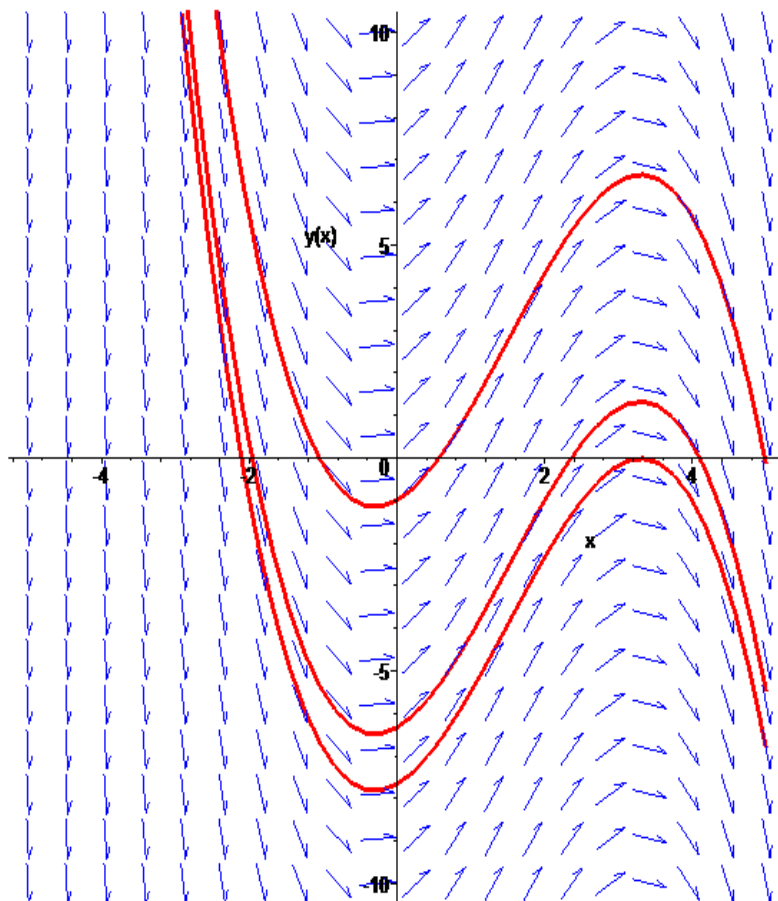
> with(DEtools):

a:=1;b:=3; DEplot(dy0,y(x),x=-5..5,y=-10..10,

[y(0)=-1,y(2)=-1,y(-2)=-1],

colour=blue,linecolor=[red,red,red]);

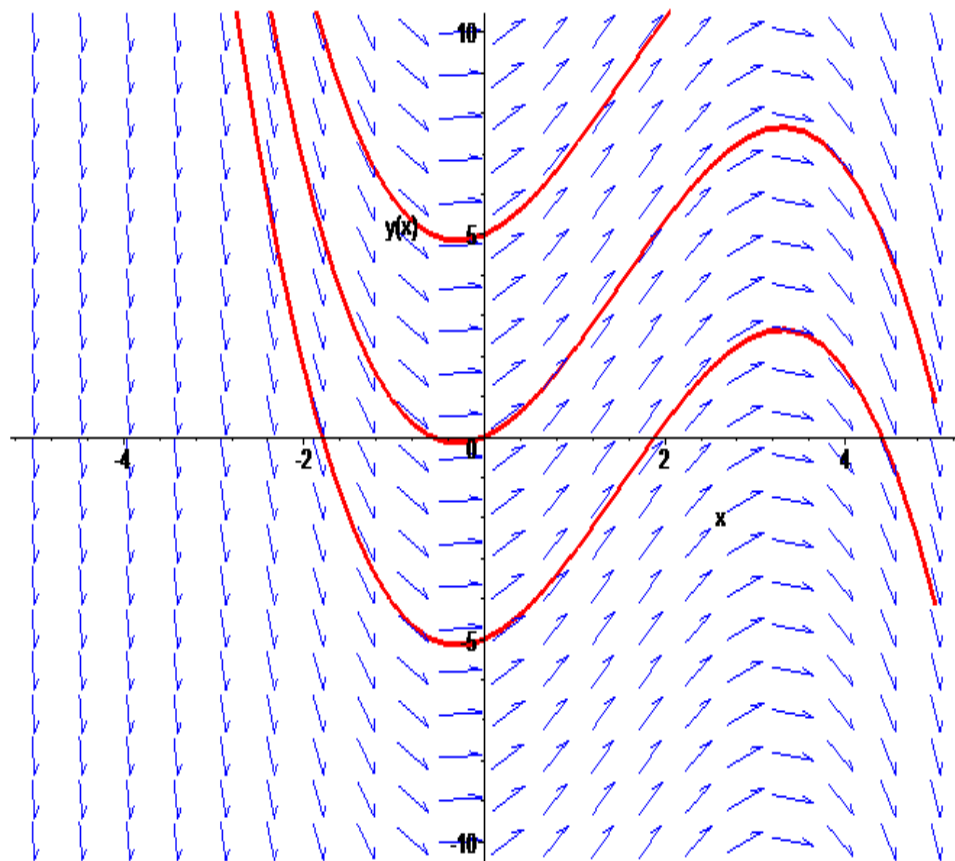
a := 1 b := 3



> with(DEtools):

```
a:=1;b:=3; DEplot(dy0,y(x),x=-5..5,y=-10..10, [y(0)=5,y(0)=0,y(0)=-5],
colour=blue,linecolor=[red,red,red]);
```

```
a := 1    b := 3
```



> restart;

dy0:=diff(y(x),x\$1)=1-x^2-3*x;

$$dy0 := \frac{d}{dx} y(x) = 1 - x^2 - 3x$$

> dsolve(dy0,y(x));

$$y(x) = -\frac{1}{3}x^3 - \frac{3}{2}x^2 + x + _C1$$

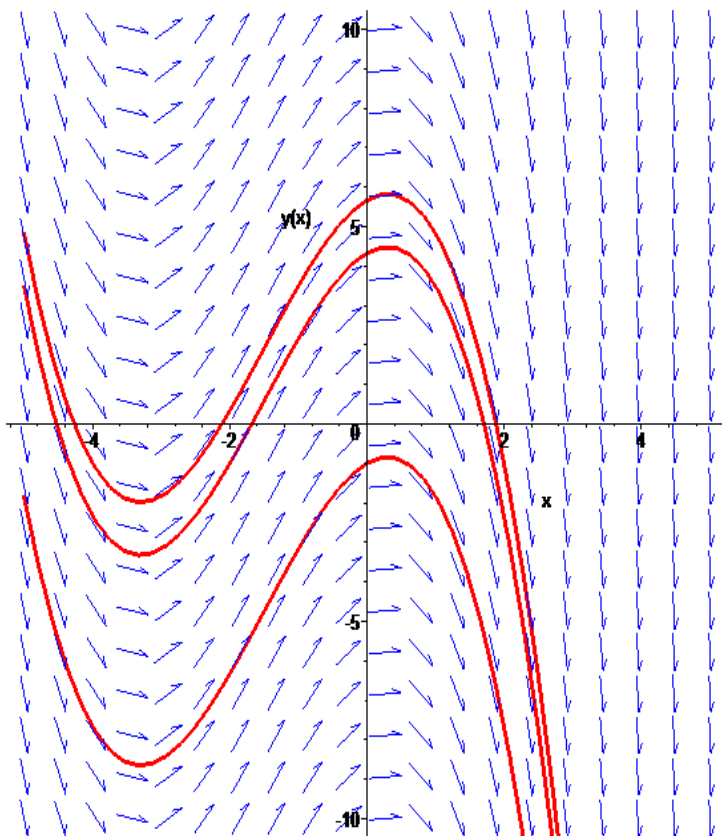
> with(DEtools):

a:=1;b:=3; DEplot(dy0,y(x),x=-5..5,y=-10..10,

[y(0)=-1,y(2)=-1,y(-2)=-1],

colour=blue,linecolor=[red,red,red]);

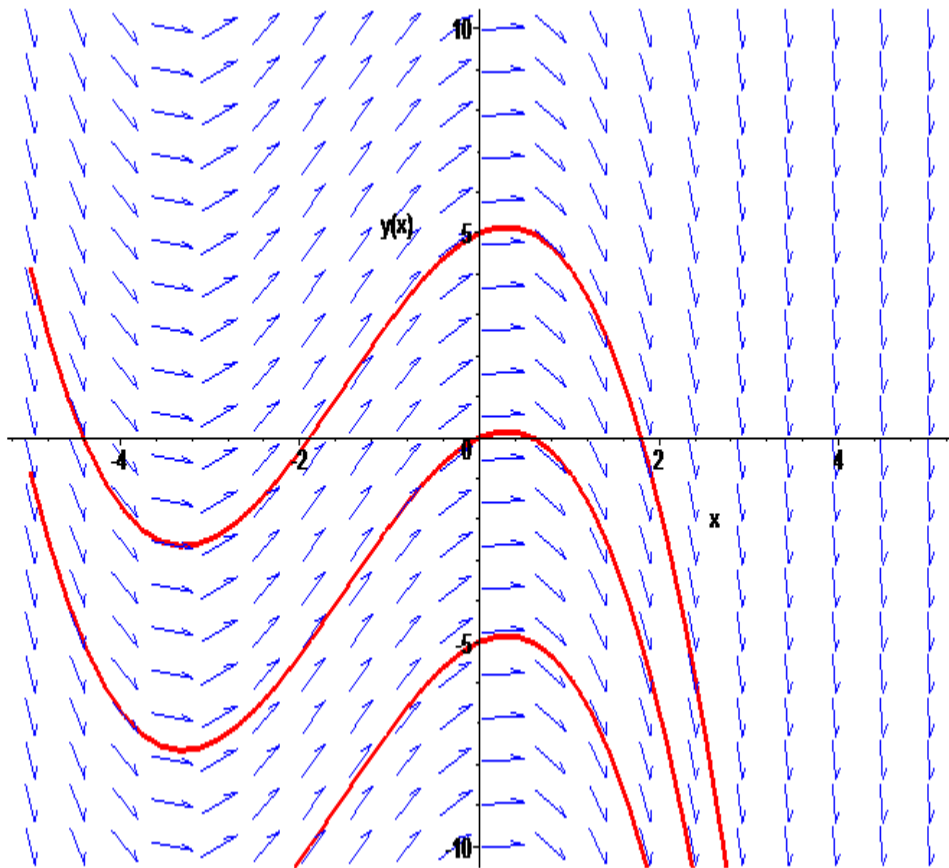
a := 1 b := 3



> with(DEtools):

a:=1;b:=3; DEplot(dy0,y(x),x=-5..5,y=-10..10,[y(0)=5,y(0)=0,y(0)=-5],
colour=blue,linecolor=[red,red,red]);

a := 1 b := 3



Заключение.

Данная работа посвящена построению функций Ляпунова для выявления свойства продолжимости всех решений некоторых нелинейных уравнений третьего порядка на полупрямую $[0; \infty)$.

В работе рассмотрены следующие нелинейные уравнения третьего порядка:

$$\ddot{x} + a\ddot{x} + \varphi(x, \dot{x})\dot{x} + f(x) = e(t)$$

$$\ddot{x} + \varphi(x, \dot{x})\ddot{x} + \psi(x, \dot{x}) + f(x) = e(t)$$

$$\ddot{x} + b\ddot{x} + \varphi(x, \dot{x}) + \psi(\dot{x}) + ax = e(t).$$

Приведенные примеры построения функций Ляпунова для выявления свойства продолжимости нелинейных уравнений третьего порядка говорят о возможности применения указанных функций не только для выяснения вопросов устойчивости, но и для выявления других свойств решений дифференциальных систем.

Литература

1. Понтрягин Л.С., Обыкновенные дифференциальные уравнения, 4-е изд., М.: Наука, -- 1974., --- 331стр.
2. Горбунов А.Д., Некоторые вопросы качественной теории обыкновенных линейных однородных дифференциальных уравнений с переменными коэффициентами. М.: Учен. зап. ун-та, 165. Математика, 7 (1954), 39--78.
3. Ла-Салль Ж., Лефшец С., Исследование устойчивости прямым методом Ляпунова, М.: Мир, 1964г.
4. Ющенко А.А., // Доклады АН БССР, т. 11, №10, 1967г.
5. Ющенко А.А., // Дифференциальные уравнения т.4 №11, 1968г.
6. Демидович Б.П., Лекции по математической теории устойчивости, М.: Наука, 1967г.
7. Барбашин Е.А., Функции Ляпунова, М.: Наука, 1970

Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»

Таразский Государственный университет им. М.Х.Дулати

Айтказина А.М.

ст. преподаватель кафедры «Прикладная математика»

Таразский Государственный университет им. М.Х.Дулати

Абдрахимова А.Н.

студентка 4 курса, спец. «Математика»

Таразский Государственный университет им. М.Х.Дулати

ОБОБЩЕННЫЕ ПОКАЗАТЕЛИ ЛЯПУНОВА

Рассмотрим характеристические показатели Ляпунова решения линейной системы

$$\lambda_{x(t)} = \overline{\lim}_{t \rightarrow +\infty} \frac{1}{t} \ln |x(t)|,$$

где $x(t) \neq 0$ - решение линейной системы обыкновенных дифференциальных уравнений

$$\dot{x} = A(t)x; \tag{ф.1}$$

Теорема Ляпунова. Пусть

$$\overline{\lim}_{t \rightarrow +\infty} \frac{1}{t} \int_0^t \|A(\tau)\| d\tau < +\infty;$$

Тогда для всякого решения $x(t) \neq 0$ системы (ф.1) характеристические показатели $\lambda_{x(t)}$ являются действительными числами (т. е. $\neq \pm\infty$).

Множество всех показателей Ляпунова ненулевых решений системы (ф.1) называют ее спектром.

Возможны следующие частные случаи:

1) Система с постоянными коэффициентами (т. е. $A(t) \equiv A(0)$). В этом случае $\lambda_i(A)$ равны действительным частям собственных значений оператора $A(0)$.

2) Система с периодическими коэффициентами (т. е. $A(t+T) \equiv A(t)$, $T > 0$). В этом случае

$$\lambda_i(A) = \frac{1}{T} \ln |\mu_i|$$

где μ_i - мультипликаторы системы (ф.1), занумерованные в порядке невозрастания их модулей (каждый берется столько раз, какова его кратность).

Рассмотрим систему

$$\frac{dy}{dt} = A(t)y, \tag{ф.2}$$

с непрерывной матрицей $A(t)$, где $t \in [t_0, +\infty)$.

Пусть $y(t)$ в общем случае комплекснозначная функция, определенная в

$$I = [t_0, +\infty);$$

\mathcal{Q} - множество положительных монотонно возрастающих непрерывных функций, определенных в I .

Определение 1: Число (или символ $-\infty$ или $+\infty$), определяемое формулой

$$\chi[y, q] = \begin{cases} \overline{\lim}_{t \rightarrow +\infty} \frac{1}{q(t)} \ln |y(t)| & \text{при } y(t) \neq 0, t \in I, \\ -\infty & \text{при } y(t) = 0, t \in I. \end{cases}$$

где $q(t) \in \mathcal{Q}$, будем называть обобщенным верхним характеристическим показателем

Ляпунова, иначе, обобщенным показателем $y(t)$ относительно $q(t)$ [1].

Допустим, что $q(t) = t^2$

Определение 2: Будем говорить, что система (ф.1) принадлежит классу $M_n(t^2)$, если любое ее ненулевое решение $y(t)$ имеет конечный обобщенный показатель относительно $q(t)$ [1].

Определение 3: Фундаментальная система решений и соответствующая фундаментальная матрица называется нормальной, если сумма ее обобщенных показателей наименьшая по сравнению с другими фундаментальными системами решений [3].

Определение 4: Обобщенные показатели нормальной фундаментальной системы решений называются обобщенными показателями Ляпунова системы (ф.2) и обозначаются [1]

$$\lambda_1, \lambda_2, \dots, \lambda_n.$$

Определение 5: Обобщенные показатели системы (ф.2) называется устойчивыми, если для любого $\varepsilon > 0$, существует $\delta > 0$, что всякий обобщенный показатель λ_j любой

возмущенный системы [1]

$$\dot{z} = A(t)z + B(t)z \quad (\text{ф.3})$$

где $\|B(t)\| \leq \delta$, $t \geq t_0$, удовлетворяет неравенству $\min_i |\lambda_{y_i} - \lambda_{i_i}| \leq \varepsilon$.

В классической теории показателей Ляпунова это понятие возникло из работы Перрона, впервые установившего, что обычные показатели могут быть неустойчивыми.

Мы, используя идею Перрона установили, что обобщенные показатели в классе

$M_n(t^2)$ тоже могут быть неустойчивыми.

Доказательство основано на следующей теореме

Теорема: В классе $M_n(t^2)$ существует устойчивая система вида (ф.2) и матрица

$B(t)$ удовлетворяющая условию $\|B(t)\| \rightarrow 0$ при $t \rightarrow +\infty$ такая, что система

$$\frac{dz}{dt} = [A(t) + B(t)]z$$

принадлежит классу $M_n(t^2)$ и является неустойчивой системой [2].

Доказательство: Чтобы доказать эту теорему, достаточно на примере показать систему, которая удовлетворяют этим условиям [1].

Пример:

$$\begin{cases} \frac{dy_1}{dt} = -2aty_1 \\ \frac{dy_2}{dt} = (2t \sin \ln t + t \cos \ln t - 4at)y_2 \end{cases} \quad (\text{ф.4})$$

Общее решение системы получим в виде (ф.5):

$$\begin{cases} y_1 = e^{-at^2 + C_1} \\ y_2 = e^{t^2 \sin \ln t - 2at^2 + C_2} \end{cases} \quad (\text{ф.5})$$

Если в качестве возмущающей матрицы выбрать матрицу

$$B(t) = \begin{pmatrix} 0 & 0 \\ e^{-at^2} & 0 \end{pmatrix},$$

то возмущенная система уравнений будет иметь следующий вид:

$$\begin{cases} \frac{dz_1}{dt} = -2atz_1 \\ \frac{dz_2}{dt} = (2t \sin \ln t + t \cos \ln t - 4at)z_2 + z_1 e^{-at^2} \end{cases} \quad (\text{ф.6})$$

Получим общее решение системы (ф.7)

$$\begin{cases} z_1 = e^{-at^2} \\ z_2 = \left(C_2 + C_1 \int e^{-t^2 \sin \ln t} dt \right) e^{t^2 \sin \ln t - 2at^2} \end{cases}$$

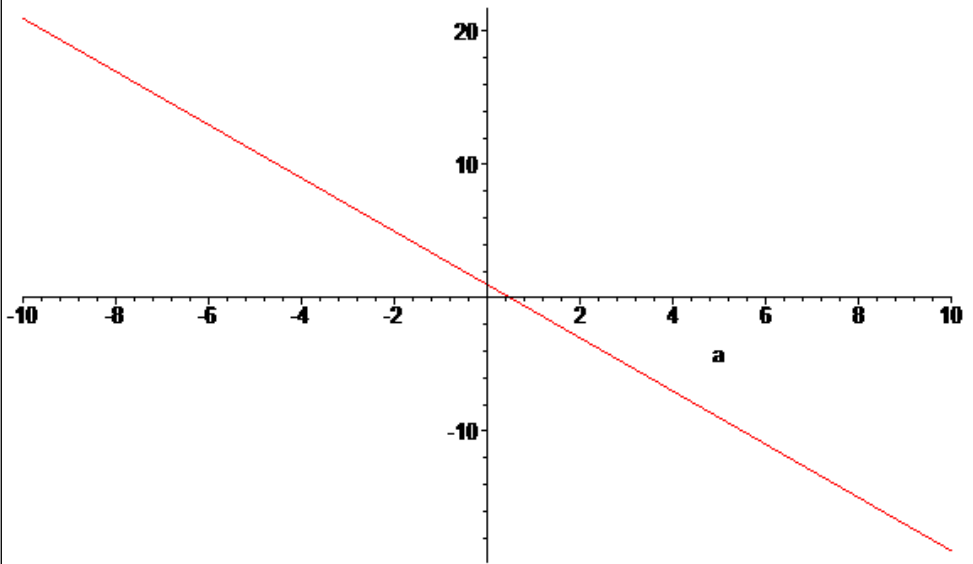
Анализируя полученные системы решений, можно сделать вывод, что решения системы (ф.4) устойчивы только в том случае, если $\tilde{N}_1 = 0$. Это условие выполняется только для тех решений, для которых $z_1(t_0) = 0$.

$$\chi[z_2] = 1 - 2a + \frac{e^{-2\pi}}{2} = 1 - 2a + \frac{1}{2e^{2\pi}}.$$

На плоскости данная функция переходит в прямую:

```
> restart;
f:=1-2*a+1/(2*exp(2*Pi));
plot(f);
```

$$f := 1 - 2a + \frac{1}{2} \frac{1}{e^{(2\pi)}}$$



Таким образом, если $z_1(t_0) \neq 0$ и параметр a такой, что имеют места неравенства

$$0 < \chi[z_2] < \frac{1}{2e^{2\pi}},$$

то решение системы

$$\frac{dz}{dt} = [A(t) + B(t)]z,$$

не устойчиво.

На основании полученных данных, можно сделать вывод, что обобщенные показатели в классе $M_n(t^2)$ являются неустойчивыми.

Литература

1. Демидович Б.П., Лекции по математической теории устойчивости. -М.: Наука, 1967.
2. Сулейменов Ж.С. Введение в теорию показателей Ляпунова. Алма-Ата, 1986.
3. Федорюк М.К. Обыкновенные дифференциальные уравнения.-М., Наука, 1985.

Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»

Таразский Государственный университет им. М.Х.Дулати

Жунисбеков С.

доктор технических наук, профессор, академик НИАРК, ректор

Таразского технического института

ОПТИМИЗАЦИЯ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ НА DELPHI

Многие задачи облачных вычислений, а также активное использование многоядерных процессоров требуют от программистов совершенствования, и внедрения многопоточных вычислений, уже на стадии разработки программ. Delphi предоставляет программисту полный доступ к возможностям программирования интерфейса Win32 и Win64 [1,с.1]. Для организации потоков предоставляется специальный класс TThread. Класс полностью упрощает программный интерфейс, предоставляя разработчику простой доступ к программированию потоков. В двух словах, все, что вам необходимо сделать, — это перекрыть виртуальный метод Execute [2-4].

Однако при попытке практической реализации можно столкнуться с целым рядом проблем, связанных со сложностью выбора потоков, правильным разбиением кода, и оценкой необходимости самих многопоточных вычислений[5-6].

Рассмотрим несколько простых циклов и алгоритмов. Разработаем программу выполняющую определенный объем вычислительной работы, точнее, расчет сумм корней из последовательно увеличивающихся индексов текущего цикла, k_1 и k_2 (рис.1.).

$$k_1 = \sum_{i=1}^{100\,000\,000} \sqrt{i},$$
$$k_2 = \sum_{i=1}^{100\,000\,000} \sqrt{i}.$$

Рассмотрим вначале реализацию этого расчета в одном последовательном потоке, а затем при создании второго дополнительного потока.

```
for I := 1 to 100000000 do
k1:=k1+sqrt(i);

for I := 1 to 100000000 do
k2:=k2+sqrt(i);
```

Рассмотрим вначале реализацию этого расчета в одном последовательном потоке.

```
procedure TForm1.Button1Click(Sender: TObject);
var t : longint;
    I: Integer;
begin
t := GetTickCount;           // Начало
k1:=0;      k2:=0;
memo1.Clear;  memo2.Clear;

for I := 1 to 100 000 000 do    k1:=k1+sqrt(i);
for I := 1 to 100 000 000 do    k2:=k2+sqrt(i);

memo1.Text:=floattostr(k1);    memo2.Text:=floattostr(k2);

t := t - GetTickCount ;
statusbar1.Panels.Items[1].Text:= 'Время работы: '+floattostr(t/1000)+ ' сек.';
label1.Caption:=statusbar1.Panels.Items[1].Text;
end;
```



Рисунок 1 - Окно программы (вып. алгоритмов в один поток).

Для создания второго потока существует несколько вариантов, принципиально отличающихся друг от друга, и как результат мы получим разное время выполнения. Рассмотрим каждый из них в отдельности, реализуем его, и оценим эффективность.

В первом случае: Создаем дополнительный поток и обращаемся к нему напрямую из алгоритма.

code: Delphi

```
Potok2:=TPotok2.Create(true);
Potok2.FreeOnTerminate:=true;
Potok2.Resume;
for I := 1 to 100000000 do
k1:=k1+exp(1/3*ln(i));
```

Причем создание можно задать как в самом алгоритме, так и в самом начале при создании формы. Причем в обоих случаях необходимо избегать последовательного расположения алгоритмов – которые мы бы хотели разделить. А значит в начале, надо запустить все доступные потоки, а уже затем основной поток в цикле.

code: Delphi

```
procedure TPotok2.Execute;
var i:integer;
begin
for I := 1 to 100000000 do
k2:=k2+sqrt(i);
end;
```

```
procedure TForm1.Button2Click(Sender: TObject);
var t : longint;
    I: Integer;
begin
t := GetTickCount; // Начало
k1:=0;
k2:=0;
memo3.Clear;
memo4.Clear;
Potok2.Resume;
for I := 1 to 100000000 do
k1:=k1+sqrt(i);
while not(potok2.Finished) do ;
memo3.Text:=floattostr(k1);
```

```
t := t - GetTickCount ;
statusbar1.Panels.Items[1].Text:= 'Время работы: '+floattostr(t/1000)+ ' сек.';
label2.Caption:=statusbar1.Panels.Items[1].Text;
end;
```

Два параллельных потока с вычислениями, независимых от основного потока в котором находится запущенная программа, создадим с помощью следующего алгоритма:

code: Delphi

```
procedure Tpotok1.Execute; var i:integer;
begin
for I := 1 to 100000000 do k1:=k1+sqrt(i);
end;
```

```
procedure TPotok2.Execute; var i:integer;
begin
for I := 1 to 100000000 do k2:=k2+sqrt(i);
end;
```

```
procedure TForm1.Button2Click(Sender: TObject);
var t : longint;
I: Integer;
begin
t := GetTickCount; // Начало
k1:=0;
k2:=0;
memo3.Clear;
memo4.Clear;
```

```
Potok1:=TPotok1.Create(true); Potok1.FreeOnTerminate:=true;
Potok2:=TPotok2.Create(true); Potok2.FreeOnTerminate:=true;
Potok1.Resume;
Potok2.Resume;
```

```
while not(potok1.Finished)or not(potok2.Finished) do ;
memo3.Text:=floattostr(k1);
memo4.Text:=floattostr(k2);
t := t - GetTickCount ;
statusbar1.Panels.Items[1].Text:= 'Время работы: '+floattostr(t/1000)+ ' сек.';
label2.Caption:=statusbar1.Panels.Items[1].Text;
end;
```

Все полученные интервалы работы разных алгоритмов создания потоков и расчетов занесем в таблицу.

№	Алгоритмы	Время выполнения (сек.)				
		1 поток	2 потока			
			Создание потока в цикле	Создание потока при создании формы	Два исполняемых потока независимых от основного потока при создании формы	
в цикле	создании формы	в цикле	создании формы			
1	Квадратный корень	4,15	2,106	2,122	3,634	3,588
2	Кубический корень (экспонента и натуральный логарифм)	25,413	13,167	13,213	18,783	19,391
3	Возведение в квадрат	1,825	0,453	0,484	2,886	3,51
4	Синус	11,95	6,162	6,209	9,11	9,219

Таким образом, получаем улучшение алгоритмов, причем создание 2-х потоков будет отнимать на 1,5 - 5 секунд больше, чем создание 1-го параллельного потока. Во всех случаях мы получаем уменьшение времени работы программы. А значит использование распараллеливания достаточно эффективно, при больших расчетах, и только в третьем расчете, при создании 2-х потоков, алгоритм оказался неэффективным и на создание потоков было затрачено дополнительно более 1,5 секунды. Выбор алгоритма всегда ложится только на разработчика программы, а данные исследования показывают простоту распараллеливания и необходимость их реализации во всех программах, для использования на современных многоядерных процессорах.

Литература.

1. TThread в Delphi. /Учебник по Delphi для профессионалов/
<http://rusdir.blogspot.com/2010/02/tthread-delphi.html>
2. Антонов А.С. Введение в параллельные вычисления.-Методическое пособие.
-Москва, 2002г. -69с.
3. Создание потоков средствами класса TThread
http://www.codingrus.ru/readarticle.php?article_id=1999
4. Первые шаги с TThread в Delphi
http://www.codingrus.ru/readarticle.php?article_id=1999
5. Аппаратные Технологии Многоядерных Процессоров
http://www.fistpgtu.ru/index.php?option=com_content&task=view&id=213&Itemid=2
6. Заплавный А.Г. Проблема эффективности применения хэш-функции MD5 с учетом современных вычислительных возможностей и параллельных вычислений.
<http://stavkombez.ru/conf/category/section7/>

Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»
Таразский Государственный университет им. М.Х.Дулати

Жунисбеков С.

доктор технических наук, профессор, академик НИАРК, ректор
Таразского технического института

КОМПЬЮТЕРНОЕ РЕШЕНИЕ И АНАЛИЗ СИСТЕМ ЛИНЕЙНЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ПЕРВОГО ПОРЯДКА

Решение многих научных и технических задач приводит к построению и интегрированию дифференциальных уравнений. В таких задачах требуется установить зависимость между переменными величинами некоторого физического, химического или другого случайного процесса и др.

При этом руководствуются следующим алгоритмом:

1. Составляют систему дифференциальных уравнений на основе условия задачи.
2. Определяют тип полученного уравнения и выбирают метод решения.
3. Находят общее решение.
4. Получают частное решение, удовлетворяющее начальным условиям.
5. Исследуют полученное решение.
6. Находят численные значения искомых величин.

Составление системы дифференциальных уравнений по условию научной или технической задачи состоит в определении математической зависимости между переменными величинами и их приращениями, в нахождении выражения для производной [1,с.1].

При составлении дифференциальных уравнений используются соответственно геометрический или механический смысл производной. Кроме того в зависимости от условия задачи, применяются соответствующие законы физики, механики, химии и других наук.

Получается, что существует большое количество разнообразных задач приводящихся к решению системы дифференциальных уравнений.

Для определенности будем рассматривать системы вида [2,с.1]:

$$\begin{cases} \frac{dx}{dt} = a_1x + b_1y + c_1 \\ \frac{dy}{dt} = a_2x + b_2y + c_2 \end{cases} \quad (\text{ф.1})$$

где $a_1, b_1, c_1, a_2, b_2, c_2$ - некоторые действительные числа.

Решением такой системы является пара функций $x(t)$ и $y(t)$, обращающая

в тождества оба уравнения системы (ф.1). Для решения системы можно применить:

1. Метод Лагранжа, вариации произвольных постоянных [3,с.1].
2. Метод интегрируемых комбинаций [4,с.436].
3. Метод Эйлера, с использованием характеристического уравнения и собственных значений [5,с.1].

В нашей работе мы будем рассматривать метод Эйлера, решения системы дифференциальных уравнений с помощью характеристического уравнения, всвязи с малой распространенностью этого метода.

Также рассмотрим методы моделирования случайных векторов в рамках многомерных распределений и рамках метода Эйлера, для моделирования случайных процессов, заданных на конечном интервале времени. Однако при формировании реализаций большой длины эти методы, как было отмечено [6,с.1], требуют большого количества вычислений и трудоемкой подготовительной работы, что затрудняет их практическое использование.

Цель исследования: Упростить методы нахождения решений, и облегчить процесс исследования и анализа полученных решений.

Метод решения систем линейных дифференциальных уравнений первого порядка (Метод Эйлера).

Линейной однородной системой с постоянными коэффициентами называется система дифференциальных уравнений вида[7,с.1]

$$\frac{dx_i}{dt} = \sum_{k=1}^n a_{ik} x_k(t), \quad i = 1, 2, \dots, n, \quad (\text{ф.2})$$

где коэффициенты a_{ik} — постоянные, а $x_k(t)$ — искомые функции от t .

Систему (ф.2) можно коротко записать в виде одного матричного уравнения

$$\frac{dX}{dt} = AX,$$

где A - матрица коэффициентов, а X – матрица искомых функций.

Простейшая однородная система дифференциальных уравнений имеет следующий вид [5,с.1]:

$$\begin{cases} \frac{dx}{dt} = k \cdot x(t) + l \cdot y(t) \\ \frac{dy}{dt} = m \cdot x(t) + n \cdot y(t) \end{cases}$$

Собственно, почти все практические примеры такой системой и ограничиваются.

Здесь:

k, l, m, n – это числа (числовые коэффициенты). В частности, один, несколько или даже все коэффициенты могут быть нулевыми.

$x(t)$ и $y(t)$ – это неизвестные функции. В качестве независимой переменной выступает переменная t .

$\frac{dx}{dt}$ и $\frac{dy}{dt}$ – первые производные неизвестных функций $x(t)$ и $y(t)$ соответственно.

Найдем решение системы дифференциальных уравнений. Это значит, найдем такие функции $x(t)$ и $y(t)$, которые удовлетворяют и первому и второму уравнению системы. Как видите, принцип очень похож на обычные системы линейных уравнений. Только там корнями являются числа, а здесь – функции.

Найденный ответ записывают в виде общего решения системы дифференциальных уравнений:

$$\begin{cases} x(t) \\ y(t) \end{cases}$$

Для системы ДУ можно решить задачу Коши, то есть, найти частное решение системы, удовлетворяющее заданным начальным условиям. Частное решение системы тоже записывают с фигурными скобками.

Более компактно систему можно переписать так:

$$\begin{cases} x' = kx + ly \\ y' = mx + ny \end{cases}$$

Рассмотрим пример:

Найти общее решение системы дифференциальных уравнений [8,с.1]

$$\begin{cases} \frac{dx}{dt} = -2x + 5y, \\ \frac{dy}{dt} = x + 3y. \end{cases}$$

Решение: Вычислим собственные значения λ матрицы A , составленной из коэффициентов заданных уравнений:

$$\det(A - \lambda E) = \begin{vmatrix} -2 - \lambda & 5 \\ 1 & 2 - \lambda \end{vmatrix} = 0, \quad \Rightarrow \quad (-2 - \lambda)(2 - \lambda) - 5 = 0, \quad \Rightarrow$$

$$(\lambda + 2)(\lambda - 2) - 5 = 0, \quad \Rightarrow \quad \lambda^2 - 9 = 0, \quad \Rightarrow \quad \lambda_1 = 3, \lambda_2 = -3.$$

В данном примере характеристическое уравнение имеет два различных действительных корня.

Найдем собственный вектор V_1 , соответствующий собственному числу $\lambda_1 = 3$. Подставляя $\lambda_1 = 3$, получаем векторно-матричное уравнение для определения V_1 :

$$(A - \lambda E)V_1 = 0.$$

$$V_1 = (V_{11}, V_{21})^T = \begin{pmatrix} V_{11} \\ V_{21} \end{pmatrix}$$

Пусть собственный вектор V_1 имеет координаты (здесь индекс Т означает операцию транспонирования). Тогда предыдущее уравнение можно записать в виде:

$$\begin{pmatrix} -2 - 3 & 5 \\ 1 & 2 - 3 \end{pmatrix} \begin{pmatrix} V_{11} \\ V_{21} \end{pmatrix} = 0, \quad \Rightarrow \quad \begin{pmatrix} -5 & 5 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} V_{11} \\ V_{21} \end{pmatrix} = 0.$$

После перемножения матриц получаем систему двух уравнений:

$$\begin{cases} -5V_{11} + 5V_{21} = 0 \\ V_{11} - V_{21} = 0 \end{cases}$$

Оба уравнения являются линейно зависимыми. Из второго уравнения находим соотношение между координатами собственного вектора: $V_{11} = V_{21}$. Полагаем $V_{21} = 1$. Следовательно, $V_{11} = 1$. Таким образом, собственный вектор V_1 имеет координаты

$$V_1 = (1, 1)^T = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Аналогично определяем 2-ой собственный вектор V_2 , соответствующий $\lambda_2 = -3$. Пусть

$$V_2 = (V_{21}, V_{22})^T = \begin{pmatrix} V_{21} \\ V_{22} \end{pmatrix}.$$

Тогда

$$\begin{pmatrix} -2+3 & 5 \\ 1 & 2+3 \end{pmatrix} \begin{pmatrix} V_{21} \\ V_{22} \end{pmatrix} = 0 \quad \text{или} \quad \begin{pmatrix} 1 & 5 \\ 1 & 5 \end{pmatrix} \begin{pmatrix} V_{21} \\ V_{22} \end{pmatrix} = 0.$$

Получаем систему двух одинаковых уравнений:

$$\begin{cases} V_{21} + 5V_{22} = 0 \\ V_{21} + 5V_{22} = 0 \end{cases}$$

Отсюда находим координаты собственного вектора V_2 :

$$V_{21} = -5V_{22}, \quad V_{22} = 1, \quad V_{21} = -5.$$

Следовательно,

$$V_2 = (-5, 1)^T = \begin{pmatrix} -5 \\ 1 \end{pmatrix}.$$

Таким образом, система уравнений имеет два различных собственных числа и два собственных вектора. Общее решение выражается формулой

$$X(t) = \begin{pmatrix} x \\ y \end{pmatrix} = C_1 \exp(3t) \begin{pmatrix} 1 \\ 1 \end{pmatrix} + C_2 \exp(-3t) \begin{pmatrix} -5 \\ 1 \end{pmatrix}$$

где C_1, C_2 – произвольные числа.

Ответ:
$$\begin{cases} x(t) = C_1 e^{3t} - 5C_2 e^{-3t} \\ y(t) = C_1 e^{3t} + C_2 e^{-3t} \end{cases}$$

Разработка программы для решения систем дифференциальных уравнений и построения фазовых портретов

Разработаем программу для автоматического решения системы ДУ, по заданным условиям задачи и начальным условиям (рис.1).

Приложение собственных значений и собственных векторов при моделировании случайных процессов. 2013г

Начальные данные | Собственные значения | Фазовый портрет | Решение системы | Фазовый портрет $x=x(t)$ | Фазовый портрет $y=y(t)$

$$\frac{dx}{dt} = -2x + 5y$$

$$\frac{dy}{dt} = 1x + 2y$$

Составить матрицу

Полный расчет

$$\begin{vmatrix} -2 - \lambda & 5 \\ 1 & 2 - \lambda \end{vmatrix} = 0$$

Найти собственные значения

Рисунок 1 – Исходные данные задачи.

Здесь мы можем задавать начальные коэффициенты системы дифференциальных уравнений, в зависимости от условий задачи.

Далее программа автоматически составляет характеристическое уравнение и решает его (рис.2).

Определяются собственные значения, и находятся собственные вектора. А также оба собственных вектора отображаются на координатной плоскости.

Решение системы дифференциальных уравнений находится и отображается на четвертой вкладке программы в аналитическом виде (рис.3).

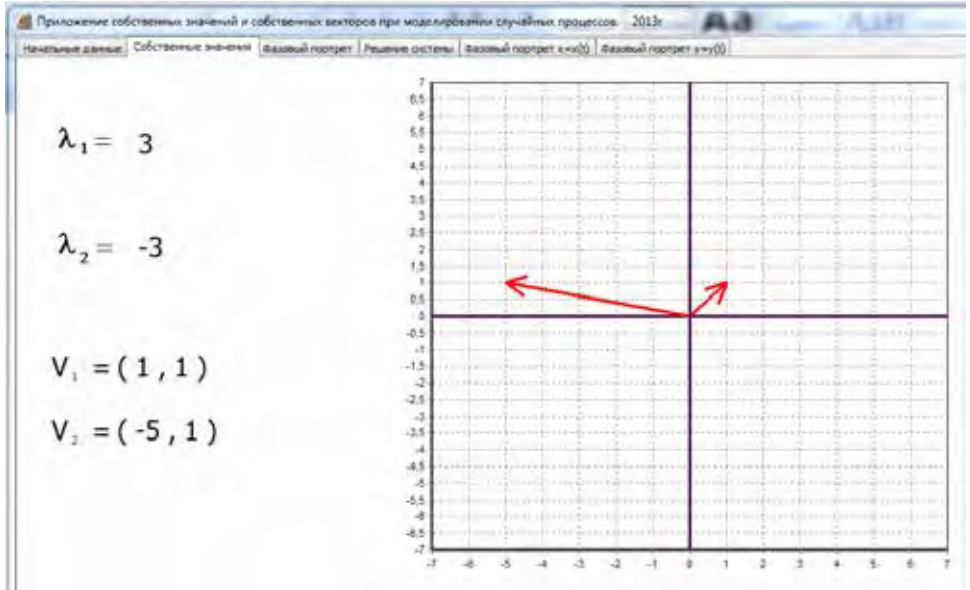


Рисунок 2 – Собственные значения и вектора.

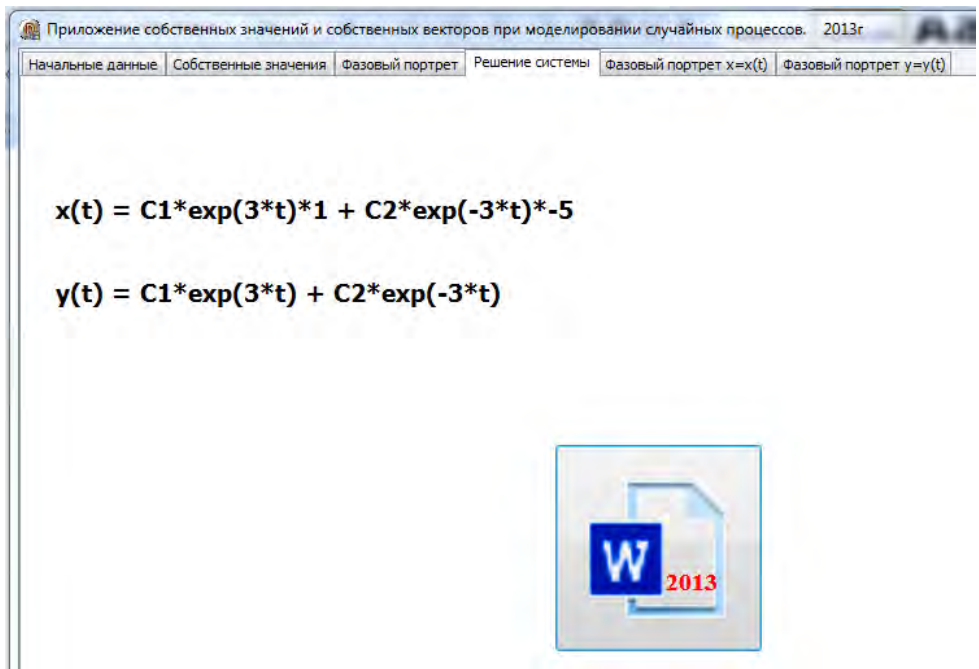


Рисунок 3 – Решение системы дифференциальных уравнений.

Построим фазовые портреты полученных решений системы дифференциальных уравнений. Варьируем переменную t в пределах от 0 до 1. А произвольные постоянные C_1 и C_2 в пределах от -10 до +10. Также зададим и некоторое частное решение и отобразим его на фазовом портрете (рис.4-7). На (рис.4) видно, что собственные вектора совпадают с асимптотическими линиями решений системы.

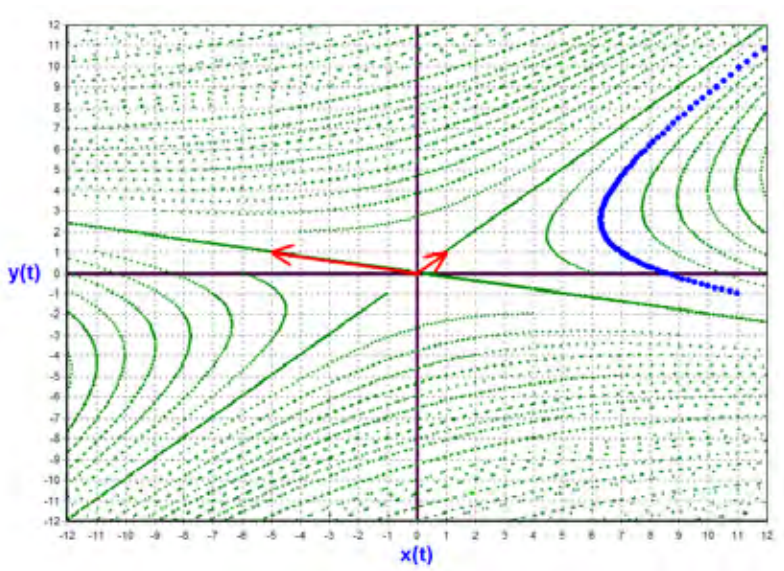


Рисунок 4 – Собственные вектора на фазовом портрете..

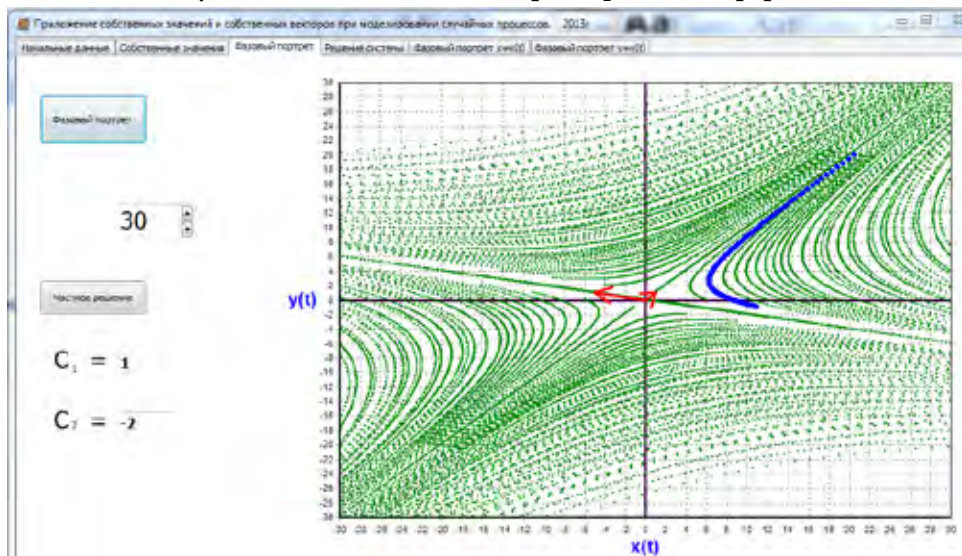


Рисунок 5 – Фазовый портрет и частное решение системы.

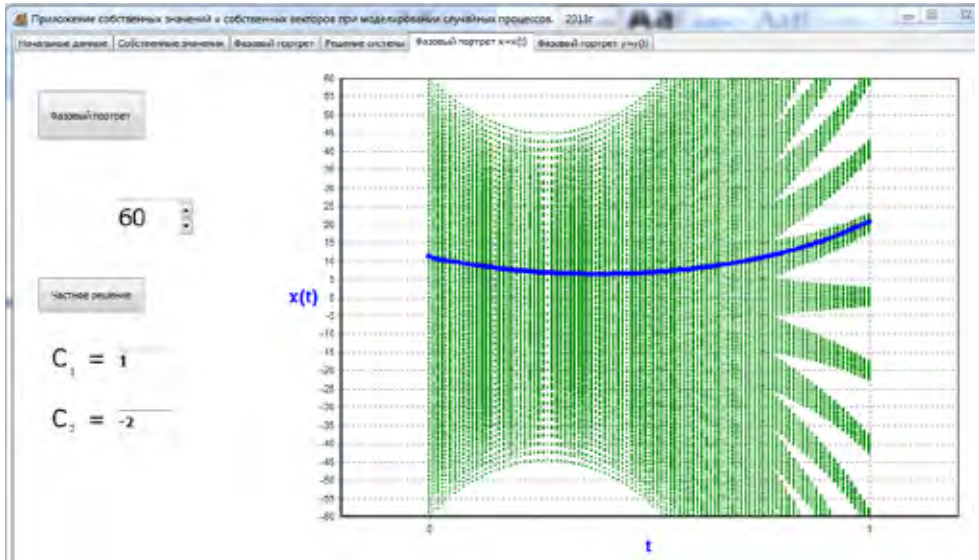


Рисунок 6 – Фазовый портрет для функции $x(t)$.

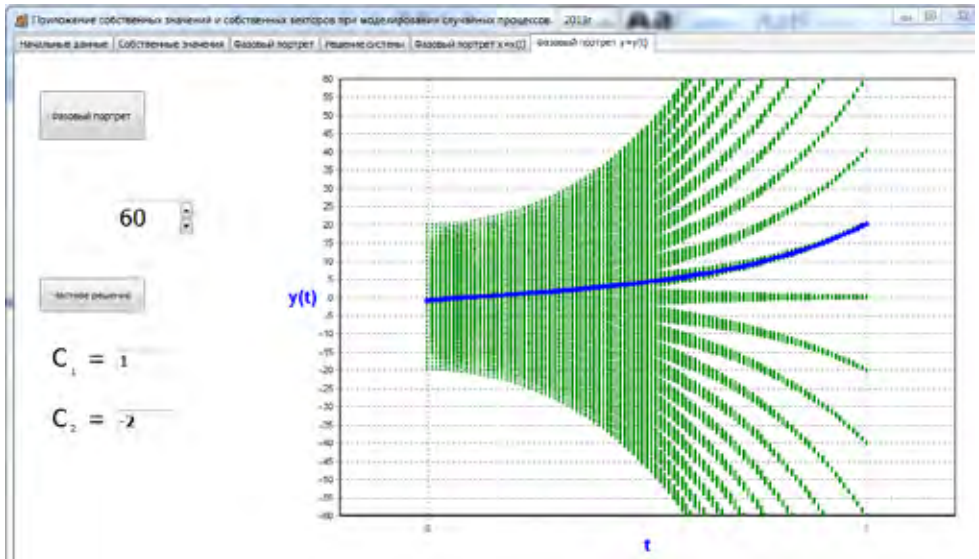


Рисунок 7 – Фазовый портрет для функции $y(t)$.

Выводы

Для рассмотренного класса задач разработана программа позволяющая

находить решение сложных систем дифференциальных уравнений и строить фазовые портреты решений. Построены фазовые портреты, наиболее полно характеризуют решения системы и позволяют проводить исследования точек сингулярности решений, стационарных и критических точек.

Разработанные алгоритмы и программа могут быть полезны при различных исследованиях, в процессе обучения, а также при решении практических задач приводимых к системам дифференциальных уравнений, и изучении асимптотической устойчивости решений.

Литература.

1. Задачи, приводящие к дифференциальным уравнениям.
<http://matica.org.ua/spravochnik-a-a-gusak-v-m-gusak/25-5-zadachi-privodyaschie-k-differentsialnim-uravneniyam>
2. Системы дифференциальных уравнений.
http://www.cleverstudents.ru/combined_differential_equations.html
3. Линейные обыкновенные дифференциальные уравнения и системы.
<http://www.pm298.ru/diffur8.php>
4. Пономарев К.К. Составление дифференциальных уравнений. –Минск, «Вышэйшая школа», 1973. -560с.
5. Емелин А. Как решить систему дифференциальных уравнений.
http://www.mathprofi.ru/sistemy_differencialnyh_uravnenij.html
6. Моделирование типовых случайных процессов.
http://www.sernam.ru/book_dm.php?id=20
7. Интегрирование однородных линейных систем с постоянными коэффициентами. Метод Эйлера.
<http://mathhelpplanet.com/static.php?p=integrirvanie-odnorodnyh-linyeinyh-sistem>
8. Метод собственных значений и собственных векторов.
<http://www.math24.ru/method-of-eigenvalues-and-eigenvectors.html>

Шевцов А.Н.

кандидат технических наук, доцент кафедры «Прикладная математика»
Таразский Государственный университет им. М.Х.Дулати

Кеулимжаева Ж.А.

магистрант 1 курса, специальности 6N0601-Математика
Таразский Государственный университет им. М.Х.Дулати

НЕКОТОРЫЕ ОСОБЕННОСТИ ПОЛИМЕРОВ

Первые упоминания о искусственных синтетических полимерах относятся к 1838 г. (поливинилиденхлорид) и 1839 г. (полистирол). Ряд полимеров был случайно получен еще в первой половине 19 века. Но в те времена химики пытались наоборот подавить полимеризацию, которая считалась побочным процессом химических реакций и сам полимер не представлял для них интереса [1].

Впервые термин «полимер» был применен Берцелиусом И. для обозначения особого вида соединений атомов - изомерии. В этой изомерии вещества (полимеры), имеющие одинаковый состав, обладали различной молекулярной массой, например этилен и бутилен, кислород и озон.

Реакции полимеризации изучались Бутлеровым А.М. изучал, а в особенности связь между строением и относительной устойчивостью молекул [1]. После создания им теории химического строения возникла химия полимеров. Наука о полимерах получила свое развитие главным образом благодаря интенсивным поискам способов синтеза каучука. В этих исследованиях принимали участие учёные многих стран, такие как: Бушарда Г., Тилден У., немецкий учёный Гарриес К., Кондаков И.Л., С.В.Лебедев и другие. Большую роль в развитии представлений о поликонденсации сыграли работы Карозерса У.

В 30-х годах были доказаны еще два способа возникновения полимерных цепочек атомов - свободно радикального и ионного.

С начала 20-х годов XX века активное развитие теория полимеров получила в работах Штаудингера Г. (теория макромолекул). До этого предполагалось, что такие биополимеры, как целлюлоза, крахмал, каучук, белки, а также некоторые синтетические полимеры, сходные с ними по свойствам (например, полиизопрен), состоят из малых молекул, обладающих необычной способностью ассоциировать в растворе в комплексы коллоидной природы благодаря не ковалентным связям (теория “малых блоков”). Однако открытие Штаудингера заставило рассматривать полимеры как качественно новый объект исследования химии и физики.

Полимеры – это химические соединения с высокой молекулярной массой, макромолекулы которых состоят из большого числа мономерных звеньев. Атомы, входящие в состав макромолекул, соединены друг с другом силами главных и координационных валентностей [1].

Свойства и основные характеристики полимеров.

Полимеры могут существовать в кристаллическом и аморфном состояниях. Необходимое условие кристаллизации - регулярность достаточно длинных

участков макромолекулы. В кристаллических полимерах возможно возникновение разнообразных надмолекулярных структур: фибрилл, сферолитов, монокристаллов, тип которых во многом определяет свойства полимерного материала. Надмолекулярные структуры в не закристаллизованных (аморфных) полимерах менее выражены, чем в кристаллических.

Получение полимеров.

Природные полимеры образуются в процессе биосинтеза в клетках живых организмов [1]. С помощью экстракции, фракционного осаждения и других методов они могут быть выделены из растительного и животного сырья.

Синтетические полимеры получают полимеризацией и поликонденсацией.

Карбоцепные полимеры обычно синтезируют полимеризацией мономеров с одной или несколькими кратными углеродными связями или мономеров, содержащих неустойчивые карбоциклические группировки.

Гетероцепные полимеры получают поликонденсацией, а также полимеризацией мономеров, содержащих кратные связи углеродо-элемента (например, $C=O$, $C=N$, $N=C=O$) или не прочные гетероциклические группировки [1].

Рассмотрим более подробно полимеризацию мономеров [2]. Способ объединения мономеров в гигантскую молекулу поясним на примере этилена C_2H_4 . Напишем структурные формулы двух молекул этилена (рис.1).

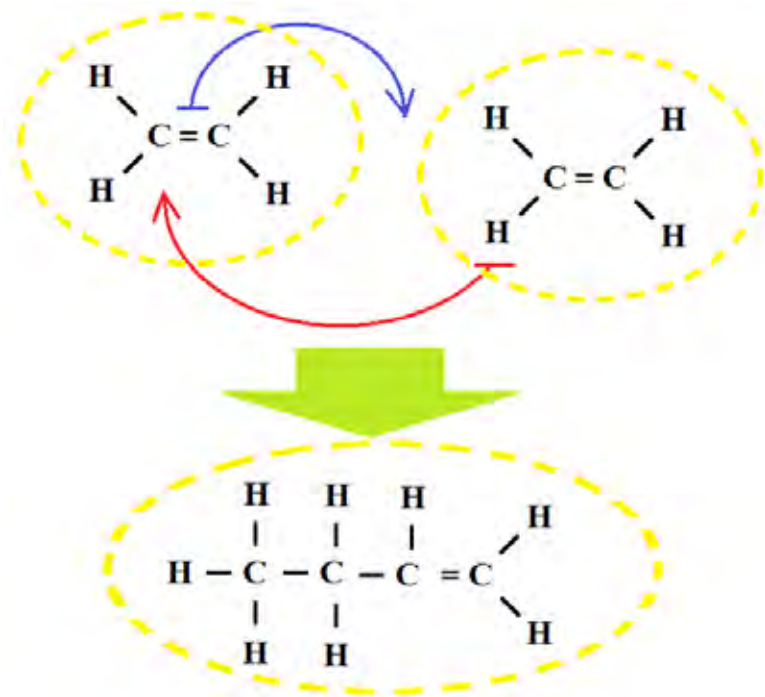


Рисунок 1 – Образование макромолекулы полимера.

Здесь атом водорода перемещается из одной молекулы в другую, в результате в этой молекуле вместо двойной связи появилась свободная одинарная связь (рис.1). Свободная связь появилась и у первой молекулы, из которой ушел водород. Поэтому эти две молекулы могут соединиться друг с другом. Такая молекула содержит уже четыре углеродных атома и одну двойную связь, как и молекула исходного этилена. Следовательно, при взаимодействии этой молекулы с еще одной молекулой этилена также может произойти перемещение атома водорода и разрыв двойной связи. Образующаяся в результате молекула будет содержать шесть атомов углерода и одну двойную связь. Таким способом можно получить последовательно молекулу с восемью, десятью и более атомами углерода. Фактически так можно получать молекулы почти любой заданной длины [2].

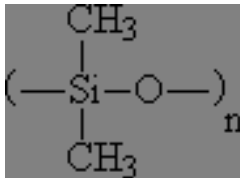
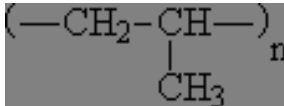
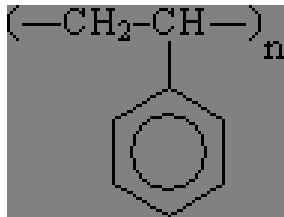
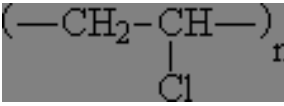
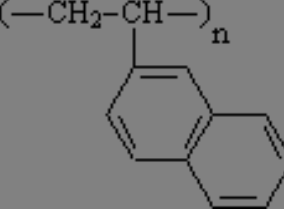
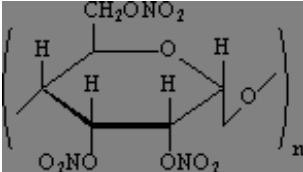
Полимеры, или макромолекулы - это очень большие молекулы, образованные связями многих молекул малого размера, которые называются составными звеньями, или мономерами. Молекулы настолько велики, что их свойства не изменяются существенным образом при добавлении или удалении нескольких таких составных звеньев.

Термин «**полимерные материалы**» объединяет три обширных группы синтетических пластиков, а именно:

- **полимеры,**
- **пластмассы,**
- **полимерные композиционные материалы (армированные пластики).**

Общее для перечисленных групп то, что их обязательной частью является **полимерная составляющая**, которая и определяет основные термомеханические и технологические свойства материала. **Полимерная составляющая представляет собой органическое высокомолекулярное вещество**, полученное в результате химической реакции между молекулами мономеров.

Таблица 1 - Статистические сегменты некоторых полимеров.

№	Полимер	Формула	Размер сегмента А (Ангстрем)	Число мономерных звеньев в сегменте
1	Полидиметилсилоксан		14,0	4,9
2	Полипропилен		21,7	8,6
3	Полистирол		20,0	7,9
4	Поливинил-хлорид		29,6	11,7
5	Поливинил-нафталин		38,7	17,4
6	Тринитро-целлюлоза		200	30

Таким образом, в зависимости от гибкости полимерной цепи её сегмент может содержать от нескольких единиц до нескольких десятков мономерных звеньев.

Современные исследования полимерных материалов и различных

многослойных композиционных конструкций определены несколькими направлениями:

- разработка наноразмерных композитных и полимерных материалов,
- создание композиционных материалов из наноструктур,
- определение механических характеристик наноразмерных композитов,
- разработка методов анализа и расчет их долговечности,
- определение устойчивости наноконструкций,
- создание математических моделей полимерных цепей на основе их молекулярной структуры.

Литература.

1.	История полимеров. stud24.ru/chemistry/istoriya-polimerov/94274-284036-page1.html
2.	http://www.e-reading-lib.org/chapter.php/83403/54/Azimov__Kratkaya_istoriya_himii._Razvitie_idei_i_predstavleniii_v_himii.html