Высшие симметрии уравнения КдФ

В.Э. Адлер. Классические интегрируемые системы, весенний семестр 2020

Уравнение КдФ

(1) $u_t = u_3 + 6\,u\,u_1$

допускает бесконечную последовательность высших симметрий нечётного порядка по производным

(2) $u_{t_{2k-1}} = f_k(u,\ ...,\ u_{2k-1}),\quad k = 1, 2\ ...,$

то есть, таких уравнений, что $[D_t,\ D_{t_{2k-1}}] = 0$.

В частности,

$u_{t_1} = f_1 = u_1$ – *классическая* симметрия, отвечающая однопараметрической группе сдвигов $x \to x + a$,

$u_{t_3} = f_2 = u_3 + 6\,u\,u_1$ – само уравнение, также классическая симметрия, отвечающая сдвигам $t \to t + a$,

$u_{t_5} = f_3 = u_5 + 10\,u\,u_3 + 20\,u_1\,u_2 + 30\,u^2\,u_1$ – первая высшая симметрия,

$u_{t_7} = f_4 = u_7 + 14\,u\,u_5 + 42\,u_1\,u_4 + 70\,u_2\,u_3 + 70\,u^2\,u_3 + 280\,u\,u_1\,u_2 + 70\,u_1^3 + 140\,u^3\,u_1$ – следующая высшая симметрия,

и т.д..

Линейное пространство конечных линейных комбинаций, порождаемое этой последовательностью как базисом, принято называть «иерархией КдФ». Точнее, это коммутативная и положительная часть иерархии; есть ещё некоммутативная и отрицательная части (ну, и дискретная часть – преобразования Бэклунда), но это мы пока пропустим.

Здесь собрано несколько способов вычисления высших симметрий (всё это будет рассказано за несколько лекций):

■ исходя из определения. Это трудоёмкий, но зато общий метод, пригодный для любого заданного эволюционного уравнения (то есть, можно либо найти высшую симметрию данного порядка, либо доказать, что её нет), кроме того, он позволяет найти всю алгебру симметрий, включая классические симметрии с зависимостью от x и t;

■ при помощи оператора рекурсии. Очень эффективный способ, если этот оператор известен. Его вывод будет дан на лекциях, а здесь мы им просто пользуемся;

■ по явным формулам из квадратичного рекуррентного соотношения. Ещё более эффективный способ, эквивалентный предыдущему;

■ по явным формулам, с использованием формального обращения преобразования Миуры; это тоже эквивалентно предыдущему;

■ при помощи мастер-симметрии. У этого метода примерно та же эффективность, что у оператора рекурсии.

Кроме того, имеется метод неопределённых коэффициентов, который вынесен в отдельную программу
06_Homogeneous polynomial equations.nb. Он применим, когда правые части как уравнения $u_t = P$, так и симметрии $u_T = Q$ – однородные многочлены. В результате, условие $[P,\ Q] = 0$ превращается в систему алгебраических уравнений на коэффициенты этих многочленов. Этот метод пригоден также для поиска законов сохранения. Его легко запрограммировать и использовать, в том числе, для поиска новых примеров.

Определим общие функции для символьных вычислений, которые нам потребуются (они уже

обсуждались в программе **02_Inversion of the Miura transformation.nb**)

In[1]:=
```
(* список динамических переменных в выражении f *)
vars[f_, u_] := Union[Cases[f, _u, {0, ∞}]]

(* df, дифференциал f *)
dif[f_] := Plus @@ (D[f, #] dd[#] & /@ vars[f, u])

(* Dx(f), Dx^k(f), полная производная по x от f *)
dx[f_] := D[f, x] + dif[f] /. dd[u[k_]] :> u[k + 1]
dx[f_, k_] := Nest[dx, f, k]

(* g*(f), оператор линеаризации g, примененный к f *)
star[g_, f_] := dif[g] /. dd[u[k_]] :> dx[f, k]

(* [f,g]=g*(f)-f*(g) *)
comm[f_, g_] := Expand[star[g, f] - star[f, g]]

(* E(f), вариационная производная или оператор Эйлера *)
vard[f_] := Expand[Plus @@ ((-1)^#[[1]] dx[D[f, #], #[[1]]] & /@ vars[f, u])]

(* порядок по производным *)
ord[f_] := Module[{v = vars[f, u]}, If[v == {}, -1, v[[-1]][[1]]]]

(* интегрирование по частям *)
int[f_] := Module[{a = 0, b = Expand[f], c, k},
  While[k = ord[b];   (* вычисляем k - порядок b *)
   And[k > 0,         (* выполняем цикл, пока выполняется условие, что k>0 и *)
    c = D[b, u[k]];   (* b линейна по старшей переменной *)
    Expand[D[c, u[k]]] == 0
   ],
   c = Integrate[c, u[k - 1]];  (* интегрируем по переменной, следующей по старшинству *)
   a = Expand[a + c];           (* пересчет a и b *)
   b = Expand[b - dx[c]]
  ];
  If[ord[b] == -1,   (* если в результате остаток не зависит от u, то это функция от x *)
   {Expand[a + Integrate[b, x]], 0},
   {a, b}
  ]
 ]
```

1 Тесты

Здесь можно посмотреть, как работают определенные выше функции.

In[10]:=
```
f = u[2] + p[u[1]]

vars[f, u]

dx[f]
dx[f, 2]

int[dx[f]]

vard[f]
vard[dx[f]]
```

Out[10]= $p[u[1]] + u[2]$

Out[11]= $\{u[1], u[2]\}$

Out[12]= $u[3] + u[2]\, p'[u[1]]$

Out[13]= $u[4] + u[3]\, p'[u[1]] + u[2]^2\, p''[u[1]]$

Out[14]= $\{p[u[1]] + u[2], 0\}$

Out[15]= $-u[2]\, p''[u[1]]$

Out[16]= 0

■ Проверка тождества Якоби

In[17]:=
```
f = u[2] + p[u[1]];
g = Sin[u[1]] + u[3];
h = u[0]^2 + u[3];
comm[f, g]
comm[g, h]
comm[h, f]
comm[f, comm[g, h]] + comm[g, comm[h, f]] + comm[h, comm[f, g]]
```

Out[20]= $\text{Sin}[u[1]]\, u[2]^2 + 3\, u[2]\, u[3]\, p''[u[1]] + u[2]^3\, p^{(3)}[u[1]]$

Out[21]= $2\,\text{Sin}[u[1]]\, u[0] - 2\,\text{Cos}[u[1]]\, u[0]\, u[1] - 6\, u[1]\, u[2] - \text{Cos}[u[1]]\, u[2]^3 - 3\,\text{Sin}[u[1]]\, u[2]\, u[3]$

Out[22]= $-2\, p[u[1]]\, u[0] + 2\, u[1]^2 + 2\, u[0]\, u[1]\, p'[u[1]] - 3\, u[2]\, u[3]\, p''[u[1]] - u[2]^3\, p^{(3)}[u[1]]$

Out[23]= 0

■ Линейные функции всегда коммутируют

In[24]:=
```
comm[u[2] + 21 u[11] + u[5], u[23] + 9 u[11] + u[0]]
comm[5 u[33] + u[101] + u[500], u[203] + 91 u[1] - 11 u[6]]
```

Out[24]= 0

Out[25]= 0

■ Уравнения типа Хопфа тоже коммутируют

In[26]:=
```
comm[a[u[0]] u[1], b[u[0]] u[1]]
```

Out[26]= 0

■ Симметрия уравнения Бюргерса

In[27]:=
```
ut2 = u[2] + 2 u[0] u[1]
ut3 = u[3] + 3 u[0] u[2] + 3 u[1]^2 + 3 u[0]^2 u[1]
Expand[star[ut2, ut3]]
comm[ut2, ut3]
```

Out[27]= $2\,u[0]\,u[1] + u[2]$

Out[28]= $3\,u[0]^2\,u[1] + 3\,u[1]^2 + 3\,u[0]\,u[2] + u[3]$

Out[29]= $18\,u[0]^2\,u[1]^2 + 12\,u[1]^3 + 6\,u[0]^3\,u[2] + 42\,u[0]\,u[1]\,u[2] + 9\,u[2]^2 + 9\,u[0]^2\,u[3] + 14\,u[1]\,u[3] + 5\,u[0]\,u[4] + u[5]$

Out[30]= 0

■ Подбор коэффициентов в симметрии уравнения Бюргерса методом неопределённых коэффициентов (продвинутые обобщения этого примера см. в отдельной программе)

In[31]:=
```
(* уравнение Бюргерса однородно относительно веса w(u_j)=1+j, вес правой части равен 3 *)
ut2 = u[2] + 2 u[0] u[1];

(* составляем многочлен общего вида веса 4, с коэффициентом 1 при u_3 *)
ut3 = u[3] + a u[0] u[2] + b u[1]^2 + c u[0]^2 u[1] + d u[0]^4;

(* вычисляем коммутатор *)
comm[ut2, ut3]

(* собираем коэффициенты *)
vars[%, u]
CoefficientList[%%, %]
Union[Flatten[%]]

(* приравниваем их 0 и решаем систему *)
Solve[% == 0]

(* находим ответ *)
ut3 = ut3 /. %[[1]]

(* проверка *)
comm[ut2, ut3]
```

Out[33]=
```
-2 d u[0]^4 u[1] - 12 d u[0]^2 u[1]^2 + 2 b u[1]^3 - 2 c u[1]^3 + 4 a u[0] u[1] u[2] -
 4 c u[0] u[1] u[2] + 6 u[2]^2 - 2 b u[2]^2 + 6 u[1] u[3] - 2 a u[1] u[3]
```

Out[34]=
```
{u[0], u[1], u[2], u[3]}
```

Out[35]=
```
{{{{0, 0}, {0, 0}, {6 - 2 b, 0}}, {{0, 6 - 2 a}, {0, 0}, {0, 0}},
  {{0, 0}, {0, 0}, {0, 0}}, {{2 b - 2 c, 0}, {0, 0}, {0, 0}}}, {{{0, 0}, {0, 0}, {0, 0}},
  {{0, 0}, {4 a - 4 c, 0}, {0, 0}}, {{0, 0}, {0, 0}, {0, 0}}, {{0, 0}, {0, 0}, {0, 0}}},
 {{{0, 0}, {0, 0}, {0, 0}}, {{0, 0}, {0, 0}, {0, 0}}, {{-12 d, 0}, {0, 0}, {0, 0}},
  {{0, 0}, {0, 0}, {0, 0}}}, {{{0, 0}, {0, 0}, {0, 0}}, {{0, 0}, {0, 0}, {0, 0}},
  {{0, 0}, {0, 0}, {0, 0}}, {{0, 0}, {0, 0}, {0, 0}}}, {{{0, 0}, {0, 0}, {0, 0}},
  {{-2 d, 0}, {0, 0}, {0, 0}}, {{0, 0}, {0, 0}, {0, 0}}, {{0, 0}, {0, 0}, {0, 0}}}}
```

Out[36]=
```
{0, 6 - 2 a, 6 - 2 b, 4 a - 4 c, 2 b - 2 c, -12 d, -2 d}
```

Out[37]=
```
{{a → 3, b → 3, c → 3, d → 0}}
```

Out[38]=
```
3 u[0]^2 u[1] + 3 u[1]^2 + 3 u[0] u[2] + u[3]
```

Out[39]=
```
0
```

2 Вычисление симметрии порядка 5 по определению

Задано уравнение $u_t = f$.

Ищем симметрию заданного порядка $u_T = g$ в общем виде, то есть, g произвольная гладкая функция от t, x, u, u_1, ..., u_n. Зависимость от t допускается, при этом

$$[D_t, D_T] = 0 \quad \Leftrightarrow \quad \partial_t(g) + g_*(f) - f_*(g) = 0$$

В этом уравнении следим только за старшей переменной (с максимальным номером).

Это даёт некоторое несложное уравнение на g.

Решаем его, уточняем вид g.

Повторяем процедуру, пересчитывая коммутатор, пока g не определится полностью.

Что значит «следим за старшей переменной» – при вычислении коммутатора на бумаге нужно вовремя остановиться и поставить многоточие вместо членов, не содержащих эту переменную. При вычислении на компьютере – не смотрим на всё выражение, а только извлекаем из него какое-то следствие, например, дифференцируя по старшей переменной.

In[40]:=
```
Clear[g]
ut = u[3] + 6 u[0] u[1];
uT = g[t, x, u[0], u[1], u[2], u[3], u[4], u[5]];
z = Expand[D[uT, t] + comm[ut, uT]];
```

Если в последней строчке убрать точку с запятой, будет выведено огромное выражение. Нам оно не нужно. Смотрим только, от чего оно зависит и дифференцируем по старшей переменной.

In[44]:=
```
vars[z, u]
D[z, u[7]]
```

Out[44]=
```
{u[0], u[1], u[2], u[3], u[4], u[5], u[6], u[7]}
```

Out[45]=
```
-3 u[6] g^(0,0,0,0,0,0,0,2)[t, x, u[0], u[1], u[2], u[3], u[4], u[5]] -
 3 u[5] g^(0,0,0,0,0,0,1,1)[t, x, u[0], u[1], u[2], u[3], u[4], u[5]] -
 3 u[4] g^(0,0,0,0,0,1,0,1)[t, x, u[0], u[1], u[2], u[3], u[4], u[5]] -
 3 u[3] g^(0,0,0,0,1,0,0,1)[t, x, u[0], u[1], u[2], u[3], u[4], u[5]] -
 3 u[2] g^(0,0,0,1,0,0,0,1)[t, x, u[0], u[1], u[2], u[3], u[4], u[5]] -
 3 u[1] g^(0,0,1,0,0,0,0,1)[t, x, u[0], u[1], u[2], u[3], u[4], u[5]] -
 3 g^(0,1,0,0,0,0,0,1)[t, x, u[0], u[1], u[2], u[3], u[4], u[5]]
```

Тут написано $-3\,D_x(\partial_5(g))$ и это должно равняться 0. Значит, $u_T = a(t)\,u_5 + g(t, x, u, \ldots, u_4)$. Чтобы экономить буквы, хвост опять обозначаем g, только от меньшего числа переменных. Переопределяем и повторяем все заново.

In[46]:=
```
Clear[a]

uT = a[t] u[5] + g[t, x, u[0], u[1], u[2], u[3], u[4]];

z = Expand[D[uT, t] + comm[ut, uT]];
vars[z, u]
D[z, u[6]]
```

Out[49]=
```
{u[0], u[1], u[2], u[3], u[4], u[5], u[6]}
```

Out[50]=
```
-3 u[5] g^(0,0,0,0,0,0,2)[t, x, u[0], u[1], u[2], u[3], u[4]] -
 3 u[4] g^(0,0,0,0,0,1,1)[t, x, u[0], u[1], u[2], u[3], u[4]] -
 3 u[3] g^(0,0,0,0,1,0,1)[t, x, u[0], u[1], u[2], u[3], u[4]] -
 3 u[2] g^(0,0,0,1,0,0,1)[t, x, u[0], u[1], u[2], u[3], u[4]] -
 3 u[1] g^(0,0,1,0,0,0,1)[t, x, u[0], u[1], u[2], u[3], u[4]] -
 3 g^(0,1,0,0,0,0,1)[t, x, u[0], u[1], u[2], u[3], u[4]]
```

Имеем $D_x(\partial_4(g)) = 0$, значит, $u_T = a(t)\,u_5 + b(t)\,u_4 + g(t, x, u, u_1, u_2, u_3)$.

In[51]:=
```
Clear[a, b]

uT = a[t] u[5] + b[t] u[4] + g[t, x, u[0], u[1], u[2], u[3]];

z = Expand[D[uT, t] + comm[ut, uT]];
vars[z, u]
Factor[D[z, u[5]]]
```

Out[54]= {u[0], u[1], u[2], u[3], u[4], u[5]}

Out[55]=
```
30 a[t] u[1] + a'[t] - 3 u[4] g^(0,0,0,0,0,2)[t, x, u[0], u[1], u[2], u[3]] -
 3 u[3] g^(0,0,0,0,1,1)[t, x, u[0], u[1], u[2], u[3]] -
 3 u[2] g^(0,0,0,1,0,1)[t, x, u[0], u[1], u[2], u[3]] -
 3 u[1] g^(0,0,1,0,0,1)[t, x, u[0], u[1], u[2], u[3]] - 3 g^(0,1,0,0,0,1)[t, x, u[0], u[1], u[2], u[3]]
```

Имеем $D_x(\partial_3(g)) = 10\,a\,u_1 + a'/3 \;\Rightarrow$

$$\partial_3(g) = 10\,a\,u + a'/3\,x + c(t) \;\Rightarrow$$

$$g = 10\,a\,u\,u_3 + (a'\,x/3 + c)\,u_3 + g(t, x, u, u_1, u_2)$$

In[56]:=
```
Clear[a, b, c]

uT = a[t] (u[5] + 10 u[0] u[3]) + b[t] u[4] + (a'[t] x/3 + c[t]) u[3] + g[t, x, u[0], u[1], u[2]];

z = Expand[D[uT, t] + comm[ut, uT]];
vars[z, u]
Factor[D[z, u[4]]]
```

Out[59]= {u[0], u[1], u[2], u[3], u[4]}

Out[60]=
```
24 b[t] u[1] + 60 a[t] u[2] + b'[t] -
 3 u[3] g^(0,0,0,0,2)[t, x, u[0], u[1], u[2]] - 3 u[2] g^(0,0,0,1,1)[t, x, u[0], u[1], u[2]] -
 3 u[1] g^(0,0,1,0,1)[t, x, u[0], u[1], u[2]] - 3 g^(0,1,0,0,1)[t, x, u[0], u[1], u[2]]
```

Имеем $D_x(\partial_2(g)) = 8\,b\;u_1 + 20\,a\,u_2 + b'/3 \;\Rightarrow$

$$\partial_2(g) = 20\,a\,u_1 + 8\,b\,u + b'\,x/3 + d(t) \;\Rightarrow$$

$$g = 20\,a\,u_1\,u_2 + 8\,b\,u\,u_2 + (b'\,x/3 + d)\,u_2 + g(t, x, u, u_1)$$

In[61]:=
```
Clear[a, b, c, d]

uT = a[t] (u[5] + 10 u[0] u[3] + 20 u[1] u[2]) + b[t] (u[4] + 8 u[0] u[2]) +
    (a'[t] x/3 + c[t]) u[3] + (b'[t] x/3 + d[t]) u[2] + g[t, x, u[0], u[1]];

z = Expand[D[uT, t] + comm[ut, uT]];
vars[z, u]
D[z, u[3]]
```

Out[64]= {u[0], u[1], u[2], u[3]}

Out[65]=
```
18 c[t] u[1] + 180 a[t] u[0] u[1] + 36 b[t] u[2] + 8 u[0] a'[t] +
 6 x u[1] a'[t] + c'[t] + 1/3 x a''[t] - 3 u[2] g^(0,0,0,2)[t, x, u[0], u[1]] -
 3 u[1] g^(0,0,1,1)[t, x, u[0], u[1]] - 3 g^(0,1,0,1)[t, x, u[0], u[1]]
```

Имеем $D_x(\partial_1(g)) = 6\,c\,u_1 + 60\,a\,u\,u_1 + 12\,b\;u_2 + 8\,a'\,u/3 + 2\,x\,a'\,u_1 + c'/3 + x\,a''/9.$

Нетрудно видеть, что это разрешимо, только если $a' = 0$ (на самом деле, если взять $a = t$, то можно

прийти к *нелокальной* мастер-симметрии из раздела 6, но сейчас мы это не будем рассматривать). Итак, пусть $a = \text{const}$, тогда

$$\partial_1(g) = 30\,a\,u^2 + 12\,b\,u_1 + 6\,c\,u + c'\,x/3 + e(t) \;\Rightarrow$$
$$g = 30\,a\,u^2\,u_1 + 6\,b\,u_1^2 + 6\,c\,u\,u_1 + (c'\,x/3 + e(t))\,u_1 + g(t, x, u)$$

In[66]:=
```
Clear[a, b, c, d, e]

uT = a (u[5] + 10 u[0] u[3] + 20 u[1] u[2] + 30 u[0]^2 u[1]) +
    b[t] (u[4] + 8 u[0] u[2] + 6 u[1]^2) + c[t] (u[3] + 6 u[0] u[1]) +
    (b'[t] x/3 + d[t]) u[2] + (c'[t] x/3 + e[t]) u[1] + g[t, x, u[0]];

z = Expand[D[uT, t] + comm[ut, uT]];
vars[z, u]
D[z, u[2]]
```

Out[69]= `{u[0], u[1], u[2]}`

Out[70]=
$$12\,d[t]\,u[1] + 96\,b[t]\,u[0]\,u[1] + 6\,u[0]\,b'[t] + 4\,x\,u[1]\,b'[t] + d'[t] + \frac{1}{3}\,x\,b''[t] - 3\,u[1]\,g^{(0,0,2)}[t, x, u[0]] - 3\,g^{(0,1,1)}[t, x, u[0]]$$

Имеем $D_x(\partial_0(g)) = 4\,d\,u_1 + 32\,b\,u\,u_1 + 2\,b'\,u + 4\,x\,b'\,u_1/3 + d'/3 + x\,b''/9.$

Отсюда получаем $b = \text{const}$ и

$$\partial_0(g) = 4\,d\,u + 16\,b\,u^2 + d'\,x/3 + f(t) \;\Rightarrow\; g = 2\,d\,u^2 + \tfrac{16}{3}\,b\,u^3 + (d'\,x/3 + f(t))\,u + g(t, x).$$

На этом этапе зависимость симметрия от u_n определена, остались только коэффициенты. Выведем коммутатор полностью – от него уже почти ничего не осталось, и соберем коэффициенты при u.

In[71]:=
```
Clear[a, b, c, d, e, f]

uT = a (u[5] + 10 u[0] u[3] + 20 u[1] u[2] + 30 u[0]^2 u[1]) +
    b (u[4] + 8 u[0] u[2] + 6 u[1]^2 + 16/3 u[0]^3) + c[t] (u[3] + 6 u[0] u[1]) +
    d[t] (u[2] + 2 u[0]^2) + (c'[t] x/3 + e[t]) u[1] + (d'[t] x/3 + f[t]) u[0] + g[t, x];

z = Collect[D[uT, t] + comm[ut, uT], u[_], Factor]
```

Out[73]=
$$4\,b\,u[1]^3 + u[1]\left(-12\,d[t]\,u[0]^2 - 32\,b\,u[0]^3 - 2\,u[0]\,(3\,f[t] - 2\,c'[t] + x\,d'[t]) + \frac{1}{3}\,(-18\,g[t, x] + 3\,e'[t] + x\,c''[t])\right) + \frac{1}{3}\,u[0]\,\left(3\,f'[t] + x\,d''[t] - 18\,g^{(0,1)}[t, x]\right) - g^{(0,3)}[t, x] + g^{(1,0)}[t, x]$$

Отсюда получаем $b = d = 0,\; f = 2\,c'/3,\; g = c''\,x/9 + g(t)$. Подставим это и пересчитаем.

In[74]:=
```
uT = a (u[5] + 10 u[0] u[3] + 20 u[1] u[2] + 30 u[0]^2 u[1]) + c[t] (u[3] + 6 u[0] u[1]) +
    (c'[t] x/3 + e[t]) u[1] + 2/3 c'[t] u[0] + c''[t] x/9 + g[t];

z = Collect[D[uT, t] + comm[ut, uT], u[_], Factor]
```

Out[75]=
$$\frac{1}{3}\,u[1]\,(-18\,g[t] + 3\,e'[t] - x\,c''[t]) + \frac{1}{9}\,\left(9\,g'[t] + x\,c^{(3)}[t]\right)$$

Имеем $c'' = 0,\ c = c_1\, t + c_2,\ g = c_3, e = 6\, c_3\, t + c_4$, где c_i произвольные постоянные.

In[76]:=
```
uT = a (u[5] + 10 u[0] u[3] + 20 u[1] u[2] + 30 u[0]^2 u[1]) +
    (c1 t + c2) (u[3] + 6 u[0] u[1]) + (c1 x / 3 + 6 c3 t + c4) u[1] + 2 / 3 c1 u[0] + c3;

z = Collect[D[uT, t] + comm[ut, uT], u[_], Factor]
```

Out[77]= 0

Итак, мы *доказали,* прямым вычислением, что все симметрии КдФ порядка не выше 5 представляются как линейная комбинация следующих симметрий:

In[78]:=
```
(* высшая симметрия *)
ut5 = u[5] + 10 u[0] u[3] + 20 u[1] u[2] + 30 u[0]^2 u[1];

(* само уравнение, симметрия сдвига по t *)
ut3 = u[3] + 6 u[0] u[1];

(* симметрия сдвига по x *)
ut1 = u[1];

(* симметрия растяжения *)
utS = 3 t (u[3] + 6 u[0] u[1]) + x u[1] + 2 u[0];

(* симметрия преобразования Галилея *)
utG = 6 t u[1] + 1;

(* проверка *)
Expand[D[#, t] + comm[ut, #]] & /@ {ut5, ut3, ut1, utS, utG}
```

Out[83]= {0, 0, 0, 0, 0}

---

3 Оператор рекурсии

Последовательность (2) можно строить по рекуррентным формулам

(3) $\quad u_{t_{2k+1}} = f_{k+1} = R^k(u_1),\ \ R = D_x^2 + 4\,u + 2\,u_1\,D_x^{-1},\ \ k = 0,\ 1,\ \ldots$

Вычислим симметрии до порядка 25. Для сравнения быстродействия: в программе с неопределенными коэффициентами за 0.5 сек мы досчитывали лишь до симметрии порядка 15.

In[84]:=
```
R[f_] := Expand[dx[f, 2] + 4 u[0] f + 2 u[1] int[f][[1]]]

M = 13;

Clear[f]
ClearSystemCache[];
Timing[
 f[1] = u[1];
 Do[f[j] = R[f[j - 1]], {j, 2, M}]]
```

Out[88]= {0.390625, Null}

Замечание. Функция **Timing** возвращает список, в котором первый элемент – время вычисления, второй – результат; в нашем случае результат пуст, так как цикл **Do** ничего не возвращает, только

что-то делает внутри себя. Перед обращением к Timing рекомендуется очищать вычисляемую переменную и кэш, так как иначе при повторном запуске ячейки время окажется заниженным – ленивая *Mathematica* не будет считать заново, а воспользуется сохраненными в памяти результатами (если какая-то мелкая операция выполняется слишком быстро, то для оценки её скорости создается цикл, в котором она повторяется достаточное число раз, причём перед каждым выполнением делается очистка памяти).

Посмотрим на несколько первых членов иерархии (можно заменить 6 на M, и полюбоваться на симметрию порядка 25). Для наглядности, перейдём к индексной записи.

In[89]:=
```
Do[Print["------------------"];
 Print[f[j] /. u[n_] :> u_n], {j, 1, 6}]
```

```
------------------

u_1

------------------

6 u_0 u_1 + u_3

------------------

30 u_0^2 u_1 + 20 u_1 u_2 + 10 u_0 u_3 + u_5

------------------

140 u_0^3 u_1 + 70 u_1^3 + 280 u_0 u_1 u_2 + 70 u_0^2 u_3 + 70 u_2 u_3 + 42 u_1 u_4 + 14 u_0 u_5 + u_7

------------------

630 u_0^4 u_1 + 1260 u_0 u_1^3 + 2520 u_0^2 u_1 u_2 + 1302 u_1 u_2^2 + 420 u_0^3 u_3 + 966 u_1^2 u_3 +
 1260 u_0 u_2 u_3 + 756 u_0 u_1 u_4 + 252 u_3 u_4 + 126 u_0^2 u_5 + 168 u_2 u_5 + 72 u_1 u_6 + 18 u_0 u_7 + u_9

------------------

2772 u_0^5 u_1 + 13860 u_0^2 u_1^3 + 18480 u_0^3 u_1 u_2 + 14784 u_1^3 u_2 + 28644 u_0 u_1 u_2^2 +
 2310 u_0^4 u_3 + 21252 u_0 u_1^2 u_3 + 13860 u_0^2 u_2 u_3 + 9702 u_2^2 u_3 + 7194 u_1 u_3^2 + 8316 u_0^2 u_1 u_4 +
 11484 u_1 u_2 u_4 + 5544 u_0 u_3 u_4 + 924 u_0^3 u_5 + 2838 u_1^2 u_5 + 3696 u_0 u_2 u_5 + 924 u_4 u_5 +
 1584 u_0 u_1 u_6 + 660 u_3 u_6 + 198 u_0^2 u_7 + 330 u_2 u_7 + 110 u_1 u_8 + 22 u_0 u_9 + u_11
```

Замечание. В принципе, возможно и сами вычисления делать с использованием индексов, но это не рекомендуется. Лучше делать wrapper – простое преобразование от одной формы к другой, как в приведённом выше примере, и применять её к выводу или вводу. Дело в том, что внутреннее представление выражения $u[n]$ совпадает с ним самим, а для u_n оно выглядит несколько неожиданно и это почти всегда приводит к путанице и неприятностям:

In[90]:=
```
FullForm[u[n]]
FullForm[u_n]
```

Out[90]//FullForm=
```
u[n]
```

Out[91]//FullForm=
```
Subscript[u, n]
```

Проверим, что симметрии попарно коммутируют (для больших порядков это занимает много времени, возьмем до $M = 6$)

In[92]:=
```
Table[comm[f[i], f[j]], {i, 1, 5}, {j, i + 1, 6}]
```

Out[92]=
```
{{0, 0, 0, 0, 0}, {0, 0, 0, 0}, {0, 0, 0}, {0, 0}, {0}}
```

4 Квадратичное рекуррентное соотношение

Обозначим $f_k = D_x(a_k)$, тогда рекуррентное соотношение (3) переписывается в виде

(4) $\quad a_1 = u, \quad D_x(a_{k+1}) = D_x^3(a_k) + 4\,u\,D_x(a_k) + 2\,u_1\,a_k, \quad k = 1, 2, \ldots$

Это эквивалентно одному уравнению для производящей функции:

(5) $\quad a_{xxx} + 4\,(\lambda + u)\,a_x + 2\,u_1\,a = 0, \quad a = 1/2 + a_1/(-4\,\lambda) + a_2/(-4\,\lambda)^2 + \ldots$

Его можно один раз проинтегрировать, умножив на $2\,a$, что даёт

(6) $\quad 2\,a\,a_{xx} - a_x^2 + 4\,(\lambda + u)\,a^2 = \lambda.$

Постоянная интегрирования в правой части определяется по свободному члену $a_0 = 1/2$. Точнее говоря, к правой части можно добавить произвольный ряд по λ^{-1} с постоянными коэффициентами, что отвечает произволу в выборе постоянных интегрирования в (4). Если сохранять эти постоянные, то просто к a_k будет добавляться линейная комбинация $a_{k-1}, \ldots, a_0$.
Полагая постоянные интегрирования равными 0, мы получаем базис из однородных многочленов.

Опять расписывая (6) по степеням λ, мы получаем новое рекуррентное соотношение. В отличие от (3) оно явное (не нужно использовать интегрирование по частям, только дифференцирование и арифметические операции), но зато квадратичное (приходится вычислять длинные суммы по предыдущим членам). В результате скорость вычислений остается примерно такой же. Но, переход от (3) к (6) замечателен тем, что он *доказывает* то, что все f_k являются полными производными и, следовательно, применение D_x^{-1} в (3) не встречает препятствий.

In[93]:=
```
Clear[a]
ClearSystemCache[];

Timing[
 a[0] = 1 / 2;
 Do[a[k + 1] =
   Expand[
    Sum[2 a[s] dx[a[k - s], 2] - dx[a[s]] dx[a[k - s]], {s, 0, k}] -
     Sum[a[s] a[k + 1 - s], {s, 1, k}] +
     4 u[0] Sum[a[s] a[k - s], {s, 0, k}]
   ], {k, 0, M - 1}]]

(* проверяем, что результаты совпадают *)
Table[Expand[dx[a[k]] - f[k]], {k, 1, M}]
```

Out[95]= {0.421875, Null}

Out[96]= {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}

5 Рекуррентные соотношения, основанные на преобразовании Миуры

Уравнение (6) можно переписать в виде следующей системы на ряды $v(z)$, $a(\lambda)$:

$$v_x + v^2 = z^2 - u, \quad a(\lambda)\,(v(-z) - v(z)) = z, \quad z^2 = -\lambda,$$

Уравнение для $v(z)$ нам уже знакомо – это преобразовании Миуры (см. программу **02_Inversion of the Miura transformation.nb**). Сначала из него находится ряд $v(z) = -z + V_1/(2\,z) + V_2/(2\,z)^2 + \ldots$, затем

из второго уравнения ряд a. В результате, расчётные формулы не такие громоздкие, как в предыдущем разделе, и считает вдвое быстрее, но, в общем-то, это одно и то же.

In[97]:=
```
Clear[a, V]
ClearSystemCache[];

Timing[
 V[1] = u[0];
 Do[V[n + 1] = Expand[dx[V[n]] + Sum[V[s] V[n - s], {s, 1, n - 1}]], {n, 1, 2 M - 2}];
 a[0] = 1 / 2;
 Do[a[k] = Expand[V[2 k - 1] + 2 Sum[a[k - s] V[2 s - 1], {s, 1, k - 1}]], {k, 1, M}]
]

(* проверяем, что результаты совпадают *)
Table[Expand[dx[a[k]] - f[k]], {k, 1, M}]
```

Out[99]= {0.15625, Null}

Out[100]= {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}

6 Мастер-симметрия

Мастер-симметрией для КдФ называется уравнение (оно получается применением оператора рекурсии к симметрии растяжения, найденной в разделе 1, подробнее это будет на лекциях)

$$u_\tau = g = x\,(u_3 + 6\,u\,u_1) + 4\,u_2 + 8\,u^2 + 2\,u_1\,D_x^{-1}(u).$$

С его помощью последовательность симметрий (2) строится по рекуррентным формулам

(7) $$u_{t_{2\,k+1}} = f_{k+1} = \frac{1}{2\,k-1}[g,\, f_k], \quad k = 1,\, 2,\, \ldots,$$

где, как обычно,

$$[g,\, f_k] = f_{k,*}(g) - g_*(f_k) = \partial_\tau(f_k) - \partial_{t_{2\,k-1}}(g).$$

Обратим внимание, что g содержит, кроме динамических переменных x, u, u_1, ... еще и *нелокальную* переменную – интеграл $u_{-1} = D_x^{-1}(u)$. При вычислении коммутатора её нужно дифференцировать по $t_{2\,k-1}$. Как это делать, понятно:

$$\partial_{t_{2\,k-1}}(u_{-1}) = \partial_{t_{2\,k-1}} D_x^{-1}(u) = D_x^{-1}\,\partial_{t_{2\,k-1}}(u) = D_x^{-1}(f_k) = a_k.$$

Определим расширенную версию функции star для выражений, содержащих u_{-1}.

In[101]:=
```
extstar[g_, f_] := dif[g] /. dd[u[-1]] → int[f][[1]] /. dd[u[k_]] :↦ dx[f, k]
extcomm[f_, g_] := Expand[extstar[g, f] - extstar[f, g]]
```

Задаем правую часть мастер-симметрии и считаем. Да, действительно, получаются высшие симметрии.

In[103]:=
```
g = x (u[3] + 6 u[0] u[1]) + 4 u[2] + 8 u[0]^2 + 2 u[1] u[-1];

extcomm[g, u[1]]
extcomm[g, %/3]
extcomm[g, %/5]
```

Out[104]= $6\,u[0]\,u[1] + u[3]$

Out[105]= $30\,u[0]^2\,u[1] + 20\,u[1]\,u[2] + 10\,u[0]\,u[3] + u[5]$

Out[106]= $140\,u[0]^3\,u[1] + 70\,u[1]^3 + 280\,u[0]\,u[1]\,u[2] + 70\,u[0]^2\,u[3] + 70\,u[2]\,u[3] + 42\,u[1]\,u[4] + 14\,u[0]\,u[5] + u[7]$

По скорости этот способ проигрывает оператору рекурсии:

In[107]:=
```
Clear[F]
ClearSystemCache[];

Timing[
 F[1] = u[1];
 Do[F[k + 1] = extcomm[g, F[k] / (2 k - 1)], {k, 1, M - 1}]
]

(* проверяем, что результаты совпадают *)
Table[F[k] - f[k], {k, 1, M}]
```

Out[109]= {1.89063, Null}

Out[110]= {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}