

October 12, 2026

1 The DenefLipshitz Algorithm: Deciding the Existence of Formal Power Series Solutions in Differential Algebra

Supplementary data for the October 2026 version of this article available at <https://hal.science/hal-05294349>. This worksheet relies on DifferentialAlgebra version 6.0

```
[1]: from sympy import *
      from DifferentialAlgebra import *
      init_printing ()
```

1.1 Section 2.1

```
[2]: x,q = var('x,q')
      y = indexedbase ('y')
      f = y*y[x]**2 + y - 1
      R = DifferentialRing (derivations = [x], blocks = [y,q,'(y)[oo:-1]'],
      ↪parameters = [q,'(y)[oo:-1]'], notation = 'jet')
```

The Ybar dictionary associates the derivative $y^{(k)}$ to the jet variable y_k for any k

```
[3]: Y = [y]
      Ybar = { y['(x,k)'] : 'y[k]' }
      pairs = R.DenefLipshitz ([f], Y, Ybar, q)
      pairs
```

```
[3]: [Denef_Lipshitz_Pair, Denef_Lipshitz_Pair, Denef_Lipshitz_Pair]
```

The three guaranteed pairs (they are guaranteed to have series solutions by Proposition 5). Compared to the article, the two last pairs (5) and (6) are interchanged

```
[4]: pairs[0].constraints ()
```

```
[4]: [y0 - 1 = 0, y1 = 0, y - 1 = 0]
```

```
[5]: pairs[1].constraints ()
```

```
[5]: [y0 - 1 = 0, y1 = 0, 2y2 + 1 = 0, y3 = 0, 2y4 + 1 = 0, yx2y + y - 1 = 0]
```

```
[6]: pairs[2].constraints ()
```

```
[6]:
```

$$[y_0 y_1^2 + y_0 - 1 = 0, 2y_0^2 y_2 + 1 = 0, y_x^2 y + y - 1 = 0, y_0 \neq 0, y_1 \neq 0]$$

The nonguaranteed pair (7):

```
[7]: ng_pairs = R.DenefLipshitz ([f], Y, Ybar, q, guaranteed = 'nonguaranteed')
ng_pairs
```

```
[7]: [Denef_Lipshitz_Pair]
```

```
[8]: ng_pairs[0].constraints ()
```

```
[8]: [y_0 - 1 = 0, y_1 = 0, y_2 = 0, y_3 = 0, y_4 = 0, y_5 = 0, y_6 = 0, y_x^2 y + y - 1 = 0]
```

A solution of any pair containing f must not annihilate the factor $2y_0 y_2 + y_1^2 + 1$ of the order one prolongation polynomial of f which implies that y_1 and y_2 cannot be both zero. Thus the nonguaranteed pair above does not describe any series solution.

```
[9]: factor (R.prolongate (f, Ybar, x))
```

```
[9]: (2y_0 y_2 + y_1^2 + 1) y_1
```

System (8) of the article has no solution:

```
[10]: R.DenefLipshitz ([f, y[0]], Y, Ybar, q)
```

```
[10]: []
```

1.2 Section 2.2

```
[11]: xi,q = var('xi,q')
y,x = indexedbase('y,x')
f = 8*x*y*y[xi,xi] + 8*y[xi] + 8*y**2 - 44*x**2 - x + 8
syst = [f, x[xi] - 1, x[0]]
R = DifferentialRing (derivations = [xi], blocks = [y,x,q,'(y,x)[oo:-1]'],
↳ parameters = [q,'(y,x)[oo:-1]'], notation = 'jet')
```

```
[12]: Y = [y,x]
Ybar = { y['(xi,k)'] : 'y[k]', x['(xi,k)'] : 'x[k]' }
```

```
[13]: pairs = R.DenefLipshitz (syst, Y, Ybar, q)
pairs
```

```
[13]: [Denef_Lipshitz_Pair, Denef_Lipshitz_Pair, Denef_Lipshitz_Pair]
```

The series solution of the first pair correspond to $y_0 = -1/2$ and the parameter y_3 is seen to be free:

```
[14]: pairs[0].series ()
```

```
[14]: { x : xi, y : xi^5 ( (113y_3 / 60 - 531 / 64) ) + xi^4 ( (171 / 64 - 17y_3 / 24) ) + (xi^3 y_3 / 6 - 9xi^2 / 8 - 5xi / 4 - 1 / 2) }
```

The second pair has no series solution for $y_0 = -1/7$ but has series solutions for other initial values such as $y_0 = 1/7$:

```
[15]: pairs[1].DenefLipshitz ([7*y[0] + 1])
```

```
[15]: []
```

```
[16]: pairs[1].DenefLipshitz ([7*y[0] - 1])
```

```
[16]: [Denef_Lipshitz_Pair]
```

The leading polynomial $A(q)$ has the root $q = 3$ for $y_0 = -1/7$:

```
[17]: A = pairs[1].leading_polynomial ()
A
```

```
[17]: {x : (1, 3, 2), y : (8qy_0 + 32y_0 + 8, 5, 4)}
```

```
[18]: solve (A[y][0].subs ({y[0] : -1/Integer(7)}), q)
```

```
[18]: [3]
```

1.3 Section 4.1

```
[19]: x,q = var('x,q')
y,z = indexedbase ('y,z')
f = y[x]**2 - z
g = z[x]**2 - y
R = DifferentialRing (derivations = [x], blocks = [y,z,q,'(y,z)[oo:-1]'],
↳parameters = [q,'(y,z)[oo:-1]'], notation = 'jet')
```

```
[20]: Y = [y,z]
Ybar = { y['(x,k)'] : 'y[k]', z['(x,k)'] : 'z[k]' }
```

Some prolongation polynomials of f :

```
[21]: R.prolongate (f, Ybar)
```

```
[21]: y_1^2 - z_0
```

```
[22]: [ R.prolongate (f, Ybar, x**i) for i in range (0,5) ]
```

```
[22]: [y_1^2 - z_0, 2y_1y_2 - z_1, 2y_1y_3 + 2y_2^2 - z_2, 2y_1y_4 + 6y_2y_3 - z_3, 2y_1y_5 + 8y_2y_4 + 6y_3^2 - z_4]
```

The polynomial $A(q)$:

```
[23]: pairs = R.DenefLipshitz ([f, y[1], Ne(y[2],0)], Y, Ybar, q)
pairs
```

```
[23]: [Denef_Lipshitz_Pair]
```

```
[24]: pairs[0].constraints ()
```

[24]: $[z_0 = 0, z_1 = 0, y_1 = 0, 2y_2^2 - z_2 = 0, -y_2z_3 + 3y_3z_2 = 0, -3y_2z_2z_4 + y_2z_3^2 + 12y_4z_2^2 = 0, y_x^2 - z = 0, y_2 \neq 0]$

[25]: `A = pairs[0].leading_polynomial () [y] [0]`
`A`

[25]: $2qy_2 + 8y_2$

Here are the prolongation polynomials of order $2k+2$ and $2k+3$ ($k = 1$). The $r = 1$ first coefficients $2y_1$ vanish on the constraints.

[26]: `f4 = R.prolongate (f, Ybar, x**4)`
`f4`

[26]: $2y_1y_5 + 8y_2y_4 + 6y_3^2 - z_4$

[27]: `f5 = R.prolongate (f, Ybar, x**5)`
`f5`

[27]: $2y_1y_6 + 10y_2y_5 + 20y_3y_4 - z_5$

Their first nonvanishing coefficients are equal to $A(q)$ for $q = 0$ and 1 .

[28]: `[ A.subs ({ q : i }) for i in range (0, 2) ]`

[28]: $[8y_2, 10y_2]$

1.4 Section 8

Recompute Example 2.2, extract the regular differential chain from the pair and compute normal forms

[29]: `xi,q = var('xi,q')`
`y,x = indexedbase('y,x')`
`f = 8*x*y*y[xi,xi] + 8*y[xi] + 8*y**2 - 44*x**2 - x + 8`
`syst = [f, x[xi] - 1, x[0]]`
`R = DifferentialRing (derivations = [xi], blocks = [y,x,q,'(y,x)[oo:-1]'],`
`parameters = [q,'(y,x)[oo:-1]'], notation = 'jet')`

[30]: `Y = [y,x]`
`Ybar = { y['(xi,k)'] : 'y[k]', x['(xi,k)'] : 'x[k]' }`

[31]: `pairs = R.DenefLipshitz (syst, Y, Ybar, q)`
`C = pairs[0].regular_differential_chain()`
`C.equations ()`

[31]: $[x_0, 2y_0 + 1, x_1 - 1, 4y_1 + 5, x_2, 4y_2 + 9, x_3, x_4, 136y_3 + 8y_4 - 513, -1808y_3 + 8y_5 + 7965, x_\xi - 1, 8y_{\xi,\xi}xy +$

[32]: `[ C.normal_form (y[i]) for i in range (0,5) ]`

[32]: $\left[-\frac{1}{2}, -\frac{5}{4}, -\frac{9}{4}, y_3, \frac{513}{8} - 17y_3\right]$

1.5 Section 13.1

```
[33]: tau,q,a,b = var('tau,q,a,b')
      t,y,z = indexedbase('t,y,z')
      eq_31 = a*t*y[tau] - (t + 1)*y + 1
      eq_32 = t*z[tau] - (b - 1)*z
      R = DifferentialRing (derivations = [tau], blocks = [y,z,t,q,'(t,y,z)[oo:
      ↪-1]',a,b], parameters = [q,a,b,'(t,y,z)[oo:-1]'], notation = 'jet')
```

```
[34]: Y = [t,y,z]
      Ybar = { y['(tau,k)'] : 'y[k]', z['(tau,k)'] : 'z[k]', t['(tau,k)'] : 't[k]',
      ↪ }
```

```
[35]: eqns = [eq_31, eq_32, a*b - 1, t[tau] - 1, t[0]]
      pairs = R.DenefLipshitz (eqns, Y, Ybar, q)
      pairs
```

```
[35]: [Denef_Lipshitz_Pair]
```

```
[36]: pairs[0].series ()
```

```
[36]:  $\left\{ t : \tau, y : \frac{b^4 \tau^4}{b^4 - 10b^3 + 35b^2 - 50b + 24} - \frac{b^3 \tau^3}{b^3 - 6b^2 + 11b - 6} + \frac{b^2 \tau^2}{b^2 - 3b + 2} - \frac{b\tau}{b - 1} + 1, z : 0 \right\}$ 
```

1.6 Section 13.2

```
[37]: tau,q = var('tau,q')
      v,t = indexedbase('v,t')
      g = 4*v*v[tau]**2 + 9*t*(v - 1)
      R = DifferentialRing (derivations = [tau], blocks = [v,t,q,'(v,t)[oo:-1]'],
      ↪parameters = [q,'(v,t)[oo:-1]'], notation = 'jet')
```

```
[38]: Y = [v,t]
      Ybar = { v['(tau,k)'] : 'v[k]', t['(tau,k)'] : 't[k]', }
```

```
[39]: pairs = R.DenefLipshitz ([g, v[0], t[tau] - 1, t[0]], Y, Ybar, q, attributes =
      ↪['differential', 'primitive'])
      pairs
```

```
[39]: [Denef_Lipshitz_Pair]
```

The constraints give the value for v_1 :

```
[40]: pairs[0].constraints ()
```

```
[40]:  $[t_0 = 0, v_0 = 0, t_1 - 1 = 0, 4v_1^3 - 9 = 0, t_2 = 0, 10v_1v_2 + 9 = 0, t_3 = 0, 350v_3 + 81 = 0, t_4 = 0, 1750v_1^2v_4 + 621$ 
```

```
[41]: vbar = pairs[0].series ()
      vbar
```

```
[41]:
```

$$\left\{ t : \tau, v : -\frac{23\tau^4 v_1}{3500} - \frac{27\tau^3}{700} - \frac{\tau^2 v_1^2}{5} + \tau v_1 \right\}$$

Perform the change of coordinate over the series and check it annihilates f :

```
[42]: x = var('x')
      y = indexedbase('y')
      Rbar = DifferentialRing (derivations = [x], blocks = [y], notation = 'jet')
      f = y*y[x]**2 + y - 1
```

```
[43]: ybar = vbar[v].subs ({tau : x**(2/Integer(3)), v[1] : (9/Integer(4))**(1/
      ↪ Integer(3))})
      ybar
```

$$-\frac{23\sqrt[3]{2} \cdot 3^{\frac{2}{3}} x^{\frac{8}{3}}}{7000} - \frac{3 \cdot 2^{\frac{2}{3}} \sqrt[3]{3} x^{\frac{4}{3}}}{20} + \frac{\sqrt[3]{2} \cdot 3^{\frac{2}{3}} x^{\frac{2}{3}}}{2} - \frac{27x^2}{700}$$

The result is actually 0 modulo x^4 :

```
[44]: collect (expand (Rbar.evaluate (f, {y : ybar })).doit()), x)
```

$$-\frac{4761 \cdot 2^{\frac{2}{3}} \sqrt[3]{3} x^{\frac{16}{3}}}{214375000} - \frac{7199 \sqrt[3]{2} \cdot 3^{\frac{2}{3}} x^{\frac{14}{3}}}{34300000} - \frac{6897 \cdot 2^{\frac{2}{3}} \sqrt[3]{3} x^{\frac{10}{3}}}{2450000} + \frac{947 \sqrt[3]{2} \cdot 3^{\frac{2}{3}} x^{\frac{8}{3}}}{122500} - \frac{12167x^6}{2679687500} - \frac{31371x^4}{17150000}$$

```
[ ]:
```