

# FORMAL POWER SERIES IN MAXIMA

An informal presentation

Bertrand Teguia Tabuguia

February 2021

## 1 Introduction

Would you like to know how the computer finds the power series formula of  $f(z) := \exp(z) + \cos(z)$  like a mathematician? What if I tell you that it computes a differential equation for  $f$ , converts that differential equation into a recurrence equation and solves that recurrence equation for the power series coefficients? Of course, some other steps should also be considered, but I will not bother you with many details. Let's present how this works for  $f(z) := \exp(z) + \cos(z)$ .

First, I load my package FPS (Formal Power Series) which contains all the necessary functions.

```
(% i1) batchload(FPS)$
```

Version 1.0

Copyright (c) May 2020, Bertrand Teguia Tabuguia

<https://www.bertrandteguia.com>

bteguia@mathematik.uni-kassel.de

University of Kassel, Germany

Yeah! I completed this package last year with my Ph.D. thesis which is the main reference for the mathematical background behind the algorithms implemented. Nevertheless, the code got better and better since June 2020 when I started the Maple version of the package. By the way, I made a demonstration in the Maple Conference 2020. Now, I would say it is "almost-surely" (I don't know the measure) "perfect". For deep understanding, I recommend (mathematicians) to read my Ph.D. thesis or the papers

1. Teguia Tabuguia, Bertrand. A variant of van Hoeij's algorithm to compute hypergeometric term solutions of holonomic recurrence equations

2. Tegua Tabugua, Bertrand and Koepf, Wolfram. Symbolic computation of hypergeometric type power series

that will appear soon. . .

Let us come back to our example. First, we compute a differential equation.

(% i2) `f:exp(z)+cos(z)$`

(% i3) `DE:HolonomicDE(f,F(z));`

$$\frac{d^3}{dz^3} F(z) - \frac{d^2}{dz^2} F(z) + \frac{d}{dz} F(z) - F(z) = 0 \quad (\% \text{ o3})$$

Then we convert it into a recurrence equation.

(% i4) `DEtoRE(DE,F(z),a[n]);`

$$(n+1)(n+2)(n+3)a_{n+3} - (n+1)(n+2)a_{n+2} + (n+1)a_{n+1} - a_n = 0 \quad (\% \text{ o4})$$

This can also be done directly as follow.

(% i5) `RE:FindRE(f,z,a[n]);`

$$(n+1)(n+2)(n+3)a_{n+3} - (n+1)(n+2)a_{n+2} + (n+1)a_{n+1} - a_n = 0 \quad (\% \text{ o5})$$

The next step is to solve that recurrence equation. Here we use a new algorithm from my Ph.D.: `mfoldHyper`, to compute what we call  $m$ -fold hypergeometric term (sometimes called Liouvillian sequences) solutions.

(% i6) `mfoldHyper(RE,a[n]);`

$$\left[ \left[ 1, \left\{ \frac{1}{n!} \right\} \right], \left[ 2, \left\{ \frac{(-1)^n z^n}{(2n)!} \right\} \right] \right] \quad (\% \text{ o6})$$

Talking about `mfoldHyper` may need another presentation... In this presentation, I only use it. You may have recognized the coefficients of the series of  $\exp(z)$  and  $\cos(z)$ . Finally we get the representation

(% i7) `FPS(f,z,n);`

$$\left( \sum_{n=0}^{\infty} \frac{(-1)^n z^{2n}}{(2n)!} \right) + \sum_{n=0}^{\infty} \frac{z^n}{n!} \quad (\% \text{ o7})$$

Isn't this amazing?! And by this process, FPS is able to do more! Even Laurent-Puiseux series are handled, I mean series with positive, negative and fractional powers.

I will only present examples for expansions in the positive neighborhood of 0 since the general case is deduced easily from it.

(% i8) FPS(log(1+sqrt(z)+z+z^(3/2)),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{(-1)^n z^{n+1}}{n+1} \right) + \sum_{n=0}^{\infty} \frac{(-1)^n z^{\frac{n+1}{2}}}{n+1} \quad (\% \text{ o8})$$

(% i9) FPS(sin(z)/z^5+sqrt(1+z)+1/sqrt(1+z),z,n);

$$\left( \sum_{n=0}^{\infty} -\frac{(-1)^n (2n-1)!((2n+1)!-3(2n)!) z^n}{4^n n!^2 ((2n-1)!-(2n)!)} \right) + \sum_{n=0}^{\infty} \frac{(-1)^n z^{2(n-2)}}{(2n+1)!} \quad (\% \text{ o9})$$

Could this be done before? This (symbolic) approach was introduced in 1992 by Wolfram Koepf, but since  $m$ -fold hypergeometric terms could not be computed before, the algorithm was reduced to some particular cases, and the result was still amazing at that time. With the results of Marko Petkovšek and Mark van Hoeij, some improvements could be done around the 2000s. Currently the only Computer Algebra System (CAS) that uses a cutting-edge implementation of Koepf's approach on its release version is Maple; Mathematica also has an implementation but van Hoeij's algorithm seems to not be used. But guess what? None of the above computations can be obtained in Maple (see my pdf file: Formal Power Series in Maple). This is normal because the coefficients cannot be detected directly by the Maple implementation. Regarding Maxima, all these computations are new and I am happy to contribute in making it among the top CASs for computing power series. Therefore there is no need to compare my implementation with the built-in Maxima command 'powerseries'. Let's move on and present more examples.

## 2 Hypergeometric type power series

You can find a definition of this subtitle in paper 2. mentioned above.

(% i10) FPS(asin(z)^2,z,n);

$$\sum_{n=0}^{\infty} \frac{4^n n!^2 z^{2(n+1)}}{(n+1)(2n+1)!} \quad (\% \text{ o10})$$

(% i11) FPS(atan(z^2),z,n);

$$\sum_{n=0}^{\infty} \frac{(-1)^n z^{2(2n+1)}}{2n+1} \quad (\% \text{ o11})$$

(% i12) FPS(exp(asinh(z)),z,n);

$$\frac{\sum_{n=0}^{\infty} \frac{(-1)^n (2n)! z^{2(n+1)}}{(n+1) 4^n n!^2}}{2} + z + 1 \quad (\% \text{ o12})$$

(% i13) FPS(integrate(erf(t),t,0,z),z,n);

$$-\frac{\sum_{n=0}^{\infty} \frac{(-1)^n z^{2n}}{(2n-1)n!}}{\sqrt{\pi}} \quad (\% \text{ o13})$$

(% i14) FPS(exp(z)-2\*exp(-z/2)\*cos(sqrt(3)\*z/2 + %pi/3),z,n);

$$3 \sum_{n=0}^{\infty} \frac{z^{3n+1}}{\left(\frac{2}{3}\right)_n \left(\frac{4}{3}\right)_n 27^n n!} \quad (\% \text{ o14})$$

Note that  $(x)_{n+1} = x(x+1)\cdots(x+n)$ . This is what we call raised factorial or Pochhammer symbol.

(% i15) FPS(z\*atan(z)-log(1+z^2)/2,z,n);

$$\frac{\sum_{n=0}^{\infty} \frac{(-1)^n z^{2(n+1)}}{(n+1)(2n+1)}}{2} \quad (\% \text{ o15})$$

(% i16) declare(k,constant)\$

I may need to find how to avoid such a specification for the user in Maxima...

(% i17) FPS(z^2/(1-z)^k,z,n);

$$\sum_{n=0}^{\infty} \frac{\binom{k}{n} z^{n+2}}{n!} \quad (\% \text{ o17})$$

(% i18) FPS(z^3\*(cos(k\*z)+sin(k\*z)),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{|k|^{2n} (-1)^n z^{2n+3}}{(2n)!} \right) + k \sum_{n=0}^{\infty} \frac{|k|^{2n} (-1)^n z^{2(n+2)}}{(2n+1)!} \quad (\% \text{ o18})$$

(% i19) FPS(exp(z)\*cos(z),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{(-1)^n 4^n z^{4n+3}}{\left(\frac{1}{4}\right)_n \left(\frac{3}{4}\right)_n (2n+1) (4n+1) (4n+3) 64^n (2n)!} \right) + \left( \sum_{n=0}^{\infty} \frac{(-1)^n 4^n z^{4n+1}}{\left(\frac{1}{4}\right)_n \left(\frac{3}{4}\right)_n (4n+1) 64^n (2n)!} \right) + \sum_{n=0}^{\infty} \frac{(-1)^n 4^n z^{4n}}{\left(\frac{1}{4}\right)_n \left(\frac{3}{4}\right)_n 64^n (2n)!} \quad (\% \text{ o19})$$

(% i20) FPS(sin(z)+z\*cos(z),z,n);

$$\sum_{n=0}^{\infty} \frac{2(n+1)(-1)^n z^{2n+1}}{(2n+1)(2n)!} \quad (\% \text{ o20})$$

(% i21) FPS((-1/2\*z + 1/6\*z^3)\*atan(z) + (-1/4\*z^2 + 1/12)\*log(z^2 + 1) + 5/12\*z^2 + 1/4,z,n);

$$\left( \sum_{n=0}^{\infty} \frac{(-1)^n z^{2(n+2)}}{4(n+1)(n+2)(2n+1)(2n+3)} \right) + \frac{1}{4} \quad (\% \text{ o21})$$

(% i22) declare(q1,constant,q2,constant)\$

(% i23) FPS(1/(q1-z^2)/(q2-z^3),z,n);

$$\begin{aligned} & \left( \sum_{n=0}^{\infty} -\frac{q1 q2^{-n-1} z^{3n+2}}{q2^2 - q1^3} \right) + \left( \sum_{n=0}^{\infty} -\frac{z^{3n+1}}{(q2^2 - q1^3) q2^n} \right) + \left( \sum_{n=0}^{\infty} \frac{z^{2n+1}}{(q2^2 - q1^3) q1^n} \right) \\ & + \left( \sum_{n=0}^{\infty} -\frac{q1^2 q2^{-n-1} z^{3n}}{q2^2 - q1^3} \right) + \sum_{n=0}^{\infty} \frac{q2 q1^{-n-1} z^{2n}}{q2^2 - q1^3} \quad (\% \text{ o23}) \end{aligned}$$

(% i24) FPS(tan(3\*atan(z^2)),z,n);

$$\left( \sum_{n=0}^{\infty} 83^{n-1} z^{4n+2} \right) + \frac{z^2}{3} \quad (\% \text{ o24})$$

(% i25) FPS((sin(z)+cos(z))^3,z,n);

$$\left( \sum_{n=0}^{\infty} \frac{3((-1)^n + (-9)^n) z^{2n+1}}{2(2n+1)(2n)!} \right) + \sum_{n=0}^{\infty} \frac{(3(-1)^n - (-9)^n) z^{2n}}{2(2n)!} \quad (\% \text{ o25})$$

(% i26) FPS(atan(z)+asin(z),z,n);

$$\sum_{n=0}^{\infty} \frac{((2n)! + (-1)^n 4^n n!^2) z^{2n+1}}{(2n+1) 4^n n!^2} \quad (\% \text{ o26})$$

(% i27) FPS(cos(z)^3+log(1+z^2),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{(4(-1)^n (2(n+1))! + n(-9)^{n+1} + (-9)^{n+1} - 3n(-1)^n - 3(-1)^n) z^{2(n+1)}}{4(n+1)(2(n+1))!} \right) + 1 \quad (\% \text{ o27})$$

(% i28) FPS(log(1+z+z^2+z^3)+sqrt(1+z^3),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{(-1)^n z^{2(n+1)}}{n+1} \right) + \left( \sum_{n=0}^{\infty} \frac{(-1)^n z^{n+1}}{n+1} \right) + \sum_{n=0}^{\infty} -\frac{(-1)^n (2n)! z^{3n}}{(2n-1) 4^n n!^2} \quad (\% \text{ o28})$$

(% i29) FPS(exp(z^2)+sinh(z)+atan(z^2),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{(-1)^n z^{4n+2}}{2n+1} \right) + \left( \sum_{n=0}^{\infty} \frac{z^{2n+1}}{(2n+1)(2n)!} \right) + \sum_{n=0}^{\infty} \frac{z^{2n}}{n!} \quad (\% \text{ o29})$$

(% i30) FPS(atan(z/(1+z^2)),z,n);

$$\sum_{n=0}^{\infty} \frac{\left( \frac{\sqrt{5}(\sqrt{5}+3)^n (-1)^n}{2^n} + \frac{(\sqrt{5}+3)^n (-1)^n}{2^n} - \frac{(\sqrt{5}-3)^n \sqrt{5}}{2^n} + \frac{(\sqrt{5}-3)^n}{2^n} \right) z^{2n+1}}{2(2n+1)} \quad (\% \text{ o30})$$

(% i31) Nmax:11\$

This variable is used to change the maximal order of the differential equation to be computed. In the following case, a differential equation of order 11 is needed.

(% i32) FPS((exp(z)-sin(z))^3,z,n);

$$\left( \sum_{n=0}^{\infty} \frac{3 \left( (-9)^{n+1} + (-1)^n \right) z^{2(n+1)+1}}{8(n+1)(2n+1)(2n+3)(2n)!} \right) + \sum_{n=0}^{\infty} -\frac{\left( 35^{\frac{n}{2}} \cos(\operatorname{atan}(2)n) + 65^{\frac{n}{2}} \sin\left(\operatorname{atan}\left(\frac{1}{2}\right)n\right) - 23^n - 3 \right) z^n}{2n!} \quad (\% \text{ o32})$$

I set the default value of *Nmax* to 10. Note that such example may increase the CPU time use in the computations.

## 2.1 Detecting identities

One of the main reason to have a direct approach is because it can simplify inputs to polynomials and constants in particular. Therefore it helps to discover identities.

(% i33) FPS(asin(z)+acos(z),z,n);

$$\frac{\pi}{2} \quad (\% \text{ o33})$$

Note that no prior knowledge is used, the result is obtained by means of symbolic computations only.

(% i34) FPS(cos(4\*acos(z)),z,n);

$$8z^4 - 8z^2 + 1 \quad (\% \text{ o34})$$

(% i35) FPS(sin(2\*asin(z))+cos(3\*acos(z)),z,n);

$$\left( \sum_{n=0}^{\infty} -\frac{2(2n)!z^{2n+1}}{(2n-1)4^n n!^2} \right) + 4z^3 - 3z \quad (\% \text{ o35})$$

(% i36) FPS(cos(2\*z)-cos(z)^2+sin(z)^2,z,n);

$$0 \quad (\% \text{ o36})$$

(% i37) f:atan((z+z^2)/(1-z^3));

$$\operatorname{atan}\left(\frac{z^2+z}{1-z^3}\right) \quad (\% \text{ o37})$$

(% i38) FPS(f,z,n);

$$\left( \sum_{n=0}^{\infty} \frac{\sin\left(\frac{\pi(n+1)}{2}\right)z^{2(n+1)}}{n+1} \right) + \sum_{n=0}^{\infty} \frac{\sin\left(\frac{\pi(n+1)}{2}\right)z^{n+1}}{n+1} \quad (\% \text{ o38})$$

In this previous example, the solution of the underlying recurrence equation contains complex numbers which reduce to real numbers when they are written in their algebraic representations.

(% i39) g:atan(z)+atan(z^2);

$$\operatorname{atan}(z^2) + \operatorname{atan}(z) \quad (\% \text{ o39})$$

(% i40) FPS(g,z,n);

$$\left( \sum_{n=0}^{\infty} \frac{\sin\left(\frac{\pi(n+1)}{2}\right)z^{2(n+1)}}{n+1} \right) + \sum_{n=0}^{\infty} \frac{\sin\left(\frac{\pi(n+1)}{2}\right)z^{n+1}}{n+1} \quad (\% \text{ o40})$$

(% i41) FPS(f-g,z,n);

$$0 \quad (\% \text{ o41})$$

(% i42) declare(y,constant)\$

(% i43) FPS(atan((z+y)/(1-z\*y)),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{(-1)^n z^{2n+1}}{2n+1} \right) + \operatorname{atan}(y) \quad (\% \text{ o43})$$

(% i44) fsqrt((1-sqrt(1-z))/z);

$$\frac{\sqrt{1 - \sqrt{1 - z}}}{\sqrt{z}} \quad (\% \text{ o44})$$

(% i45) FPS(f,z,n);

$$\frac{\sum_{n=0}^{\infty} \frac{(\frac{1}{4})_n (\frac{3}{4})_n 4^n z^n}{(2n+1)!}}{\sqrt{2}} \quad (\% \text{ o45})$$

(% i46) g:(sqrt(1+sqrt(z))-sqrt(1-sqrt(z)))/(sqrt(2\*z));

$$\frac{\sqrt{\sqrt{z} + 1} - \sqrt{1 - \sqrt{z}}}{\sqrt{2} \sqrt{z}} \quad (\% \text{ o46})$$

(% i47) FPS(g,z,n);

$$\frac{\sum_{n=0}^{\infty} \frac{(\frac{1}{4})_n (\frac{3}{4})_n 4^n z^n}{(2n+1)!}}{\sqrt{2}} \quad (\% \text{ o47})$$

(% i48) FPS(f-g,z,n);

$$0 \quad (\% \text{ o48})$$

Isn't this nice?!

## 2.2 Some Puiseux series (series with fractional powers)

(% i49) FPS(asin(z^(1/3)),z,n);

$$\sum_{n=0}^{\infty} \frac{(2n)! z^{\frac{2n+1}{3}}}{(2n+1) 4^n n!^2} \quad (\% \text{ o49})$$

(% i50) FPS(log(1+z^(1/5)),z,n);

$$\sum_{n=0}^{\infty} \frac{(-1)^n z^{\frac{n+1}{5}}}{n+1} \quad (\% \text{ o50})$$



(% i51) FPS(1/(1-z^(1/7)),z,n);

$$\sum_{n=0}^{\infty} z^{\frac{n}{7}} \quad (\% \text{ o51})$$

(% i52) FPS(cos(sqrt(z))+sin(z^(3/4)),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{(-1)^n z^{\frac{6n+3}{4}}}{(2n+1)(2n)!} \right) + \sum_{n=0}^{\infty} \frac{(-1)^n z^n}{(2n)!} \quad (\% \text{ o52})$$

(% i53) FPS(atan(sqrt(z))+asinh(z^(1/3)),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{(-1)^n z^{\frac{6n+3}{6}}}{2n+1} \right) + \sum_{n=0}^{\infty} \frac{(-1)^n (2n)! z^{\frac{4n+2}{6}}}{(2n+1) 4^n n!^2} \quad (\% \text{ o53})$$

(% i54) FPS(sqrt(sqrt(8\*z^3+1)-1)+sqrt(13\*z^4+7),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{2 \left(\frac{1}{4}\right)_n \left(\frac{3}{4}\right)_n (-8)^n 4^n z^{\frac{6n+3}{2}}}{(2n+1)(2n)!} \right) + \sum_{n=0}^{\infty} -\frac{7^{\frac{1}{2}-n} (-13)^n (2n)! z^{4n}}{(2n-1) 4^n n!^2} \quad (\% \text{ o54})$$

The sum of these two latter series can be written as a sum of four series as shown below.

(% i55) FPS(asin(z^(1/3))+log(1+z^(1/5)),z,n);

$$\left( \sum_{n=0}^{\infty} \frac{(2n)! z^{\frac{10n+5}{15}}}{(2n+1) 4^n n!^2} \right) + \left( \sum_{n=0}^{\infty} \frac{3(-1)^n z^{\frac{3n+2}{15}}}{3n+2} \right) + \left( \sum_{n=0}^{\infty} -\frac{3(-1)^n z^{\frac{3n+1}{15}}}{3n+1} \right) + \sum_{n=0}^{\infty} \frac{3(-1)^n z^{\frac{n+1}{15}}}{n+1} \quad (\% \text{ o55})$$

One can also recover the basic representation by using 'map(lambda([f],FPS(f,z,n)),.)'.

The above result is related to a linear system solved in the algorithm. When there are many possible representations, the algorithm selects a particular one.

(% i56) map(lambda([f],FPS(f,z,n)),asin(z^(1/3))+log(1+z^(1/5)));

$$\left( \sum_{n=0}^{\infty} \frac{(2n)! z^{\frac{2n+1}{3}}}{(2n+1) 4^n n!^2} \right) + \sum_{n=0}^{\infty} \frac{(-1)^n z^{\frac{n+1}{5}}}{n+1} \quad (\% \text{ o56})$$

I can provide many more examples, but let's stop here and see other kind of representations.

### 2.3 Series related to hypergeometric type series

Some formulas do not fit into the category of hypergeometric type series. However we can still connect their representations with hypergeometric type series. This is the situation for products and reciprocals of hypergeometric type series. The outputs are given as a recursive formula, by application of the Cauchy product rule or both. Our first example is the generating function of the Bernoulli numbers.

(% i57) FPS(1/(exp(z)-1),z,n);

$$\left[ \sum_{n=0}^{\infty} A_n z^n, A_n = \sum_{k=1}^n -\frac{A_{n-k}}{k!}, A_0 = 1 \right] \quad (\% \text{ o57})$$

(% i58) FPS(exp(z)\*log(1+z),z,n);

$$\sum_{n=0}^{\infty} \left( \sum_{k=0}^n \frac{(-1)^{n-k}}{k!(n-k+1)} \right) z^{n+1} \quad (\% \text{ o58})$$

(% i59) FPS(log(1+z)/exp(z),z,n);

$$\sum_{n=0}^{\infty} (-1)^n \left( \sum_{k=0}^n \frac{1}{k!(n-k+1)} \right) z^{n+1} \quad (\% \text{ o59})$$

(% i60) FPS(tan(z),z,n);

$$\left[ \sum_{n=0}^{\infty} \left( \sum_{k=0}^n \frac{A_k (-1)^{n-k}}{(2n-2k+1)!} \right) z^{2n+1}, A_k = \sum_{j=1}^k -\frac{A_{k-j} (-1)^j}{(2j)!}, A_0 = 1 \right] \quad (\% \text{ o60})$$

(% i61) FPS(1/atan(z),z,n);

$$\left[ \sum_{n=0}^{\infty} A_n z^{2n-1}, A_n = \sum_{k=1}^n -\frac{(-1)^k A_{n-k}}{2k+1}, A_0 = 1 \right] \quad (\% \text{ o61})$$

(% i62) FPS(atan(z)\*sin(z),z,n);

$$\sum_{n=0}^{\infty} (-1)^n \left( \sum_{k=0}^n \frac{1}{(2k+1)(2n-2k+1)!} \right) z^{2n+2} \quad (\% \text{ o62})$$

(% i63) FPS(atan(z)^2,z,n);

$$\sum_{n=0}^{\infty} (-1)^n \left( \sum_{k=0}^n \frac{1}{(2k+1)(2n-2k+1)} \right) z^{2n+2} \quad (\% \text{ o63})$$

## 2.4 Recursive representations from recurrence equations

In certain cases the algorithm finds a recurrence equation but does not find a representation related to hypergeometric type series. In this case it uses the recurrence equation obtained to write a recursive formula for the coefficients.

(% i64) FPS(exp(z+z^2),z,n);

$$\left[ \sum_{n=0}^{\infty} A_n z^n, A_{n+2} = \frac{A_{n+1} + 2A_n}{n+2}, n \geq 0, [A_0 = 1, A_1 = 1] \right] \quad (\% \text{ o64})$$

(% i65) FPS(sin(1+z+z^2),z,n);

$$\left[ \sum_{n=0}^{\infty} A_n z^n, A_{n+5} = \frac{-2(n+2)(n+4)A_{n+4} - A_{n+3} - 6A_{n+2} - 12A_{n+1} - 8A_n}{(n+4)(n+5)}, n \geq 0, \left[ A_0 = \sin(1), A_1 = \cos(1), A_2 = -\frac{\sin(1) - 2\cos(1)}{2}, A_3 = -\frac{6\sin(1) + \cos(1)}{6}, A_4 = -\frac{11\sin(1) + 12\cos(1)}{24} \right] \right] \quad (\% \text{ o65})$$

(% i66) FPS(atan(z)\*exp(z),z,n);

$$\left[ \sum_{n=0}^{\infty} A_n z^n, A_{n+4} = \frac{2(n+3)A_{n+3} - ((n+2)^2 + n+3)A_{n+2} + 2(n+2)A_{n+1} - A_n}{(n+3)(n+4)}, n \geq 0, \left[ A_0 = 0, A_1 = 1, A_2 = 1, A_3 = \frac{1}{6} \right] \right] \quad (\% \text{ o66})$$

Observe that this can be used to compute truncated series. My package contains the function 'Taylor' with the same syntax as the internal Maxima 'taylor' to compute Taylor polynomials.

(% i67) Taylor(sin(z),z,0,7);

$$-\frac{z^7}{5040} + \frac{z^5}{120} - \frac{z^3}{6} + z \quad (\% \text{ o67})$$

It is easy to see that for larger order expansions this approach has a better efficiency than what is internally implemented. Let us consider truncated series of arctan(z)<sup>2</sup>.

(% i68) Taylor(atan(z)^2,z,0,10);

$$\frac{563z^{10}}{1575} - \frac{44z^8}{105} + \frac{23z^6}{45} - \frac{2z^4}{3} + z^2 \quad (\% \text{ o68})$$

(% i69) `taylor(atan(z)^2,z,0,10);`

$$z^2 - \frac{2z^4}{3} + \frac{23z^6}{45} - \frac{44z^8}{105} + \frac{563z^{10}}{1575} + \dots \quad (\% \text{ o69})/T$$

(% i70) `showtime:true$`

Evaluation took 0.0000 seconds (0.0000 elapsed) using 0 bytes.

(% i71) `Taylor(atan(z)^2,z,0,1000)$`

Evaluation took 0.8900 seconds (0.8880 elapsed) using 385.493 MB.

(% i72) `taylor(atan(z)^2,z,0,1000)$`

Evaluation took 9.5630 seconds (9.5980 elapsed) using 1248.239 MB.

(% i73) `showtime:false$`

Really interesting, right?

### 3 Non-holonomic power series

I will only present three examples for this section. These are series that do not satisfy linear differential equations. The outputs are given as recursive formulas from computed algebraic ordinary differential equations. The most important aspect of this approach is that it can prove difficult identities.

(% i74) `FPS(log(1+sin(z)),z,n);`

$$\left[ \sum_{n=0}^{\infty} A_n z^n, A_{n+4} = \frac{1}{(n+2)(n+3)(n+4)} \left( -(n+2)(n+3)A_{n+3} - \left( \sum_{k=1}^n (k+1)(k+2)A_{k+2}(n-k+2)A_{n-k+2} \right) + (n+2)A_{n+2} \right), n \geq 0, \left[ A_0 = 0, A_1 = 1, A_2 = -\frac{1}{2}, A_3 = \frac{1}{6} \right] \right] \quad (\% \text{ o74})$$

(% i75) f:(1+tan(z))/(1-tan(z));

$$\frac{\tan(z) + 1}{1 - \tan(z)} \quad (\% \text{ o75})$$

(% i76) FPS(f,z,n);

$$\left[ \sum_{n=0}^{\infty} A_n z^n, A_{n+3} = \frac{2(\sum_{k=1}^n (k+1) A_{k+1} A_{n-k+1}) + 2(n+2) A_{n+2} + 4A_{n+1}}{(n+2)(n+3)}, \right. \\ \left. n \geq 0, [A_0 = 1, A_1 = 2, A_2 = 2] \right] \quad (\% \text{ o76})$$

(% i77) g:exp(2\*atanh(sin(2\*z)/(1+cos(2\*z))));

$$\% e^{2 \operatorname{atanh}\left(\frac{\sin(2z)}{\cos(2z)+1}\right)} \quad (\% \text{ o77})$$

(% i78) FPS(g,z,n);

$$\left[ \sum_{n=0}^{\infty} A_n z^n, A_{n+4} = \frac{1}{(n+2)(n+3)(n+4)} \right. \\ \left( 3 \left( \sum_{k=1}^n (k+1)(k+2) A_{k+2} (n-k+2) A_{n-k+2} \right) - \left( \sum_{k=1}^n (k+1)(k+2)(k+3) A_{k+3} A_{n-k+1} \right) \right) \\ \left. - 4 \left( \sum_{k=1}^n (k+1) A_{k+1} A_{n-k+1} \right) + 6(n+2)(n+3) A_{n+3} + 8(n+2) A_{n+2} - 24A_{n+1} \right), \\ \left. n \geq 0, \left[ A_0 = 1, A_1 = 2, A_2 = 2, A_3 = \frac{8}{3} \right] \right] \quad (\% \text{ o78})$$

Yeah, this approach does not always yield nice outputs. Although we get a different representation for the two latter expressions (this does not happen with my Maple package), the algorithm is able to prove that they are identical.

(% i79) FPS(f-g,z,n);

$$0 \quad (\% \text{ o79})$$

## 4 Conclusion

As I said, I just presented a few examples. I hope you enjoyed seeing those. My Maxima package will soon be available on the Github repository of maxima-packages. Feel free to use it and contact me if you have any question.