arXiv:2602.04531v1 [cs.SC] 4 Feb 2026

SageMath 中超几何函数的代数与算术属性

Xavier Caruso 法国国家科学研究中心；IMB，波尔多大学，塔朗斯，法国 xavier@caruso.ovh

Florian Fürnsinn 维也纳大学，数学系，维也纳，奥地利 florian.fuernsinn@univie.ac.at

摘要

我们介绍了计算机代数系统 SageMath 中用于处理超几何函数代数与算术性质的算法实现。我们研究了有理数、有限域以及 p-进数域上的超几何级数。此外，我们还实现了判定代数性、计算赋值以及在正特征下计算最小多项式的功能。

1 引言

一个 带有有理数 上参数 $\boldsymbol{\alpha}$的超几何函数(α_1，…，α_n =) $\in \mathbb{Q}^n$ 和下参数 $\boldsymbol{\beta}$：(β_1，…，β_m) $\in \mathbb{Q}^m$定义为幂级数

$$ {}_nF_m(\boldsymbol{\alpha},\boldsymbol{\beta};x) \coloneqq \sum_{k=0}^{\infty} \frac{(\alpha_1)_k \cdots (\alpha_n)_k}{(\beta_1)_k \cdots (\beta_m)_k} \cdot \frac{x^k}{k!} \in \mathbb{Q}[[x]], \qquad (1) $$

其中$(\gamma)_k$：$= \gamma(\gamma+1)\cdots(\gamma+k-1)$ 表示上升阶乘或*Pochhammer*符号。

超几何函数 ${}_nF_m(\boldsymbol{\alpha}, \boldsymbol{\beta}; x)$ 是微分方程 $\mathrm{L}(\boldsymbol{\alpha}, \boldsymbol{\beta}, x)y(x) = 0$ 的解，其中 $\vartheta = x\,\mathrm{d}\,\mathrm{dx}$ 表示，

$$ \mathcal{L}(\boldsymbol{\alpha},\boldsymbol{\beta},x) \coloneqq \big(x(\vartheta+\alpha_1)\cdots(\vartheta+\alpha_n) - \vartheta(\vartheta-\beta_1)\cdots(\vartheta-\beta_m)\big). $$

如果存在非零多项式 $P(x, y) \in K[x, y]$，使得 $f(x, y(x)) = 0$，则幂级数 $f(x) \in K[[x]]$ 称为代数的。类似地，如果存在非零微分算子 $L \in \mathbb{Q}[x]\partial$，使得 $Lf(x) = 0$，则幂级数 $f(x) \in \mathbb{Q}[[x]]$ 称为*D*有限。此外，如果它具有正的收敛半径，并且存在两个非零整数 α，$\beta \in \mathbb{Z}$，使得 $\beta f(\alpha x) \in \mathbb{Z}[[x]]$，则称其全局有界。

超几何函数显然是 D有限的，其为代数的或全局有界的参数集合也已得到完整分类。此外，只要其有定义，它们模 p （其中 $\mathbb{F}_p[[x]]$）的约化就是代数的。这些性质使其成为检验关于 D有限级数的代数性质和算术性质的猜想的理想案例。作者与 Vargas-Montoya 合作研究 D 有限级数模素数约化的伽罗瓦群 [5] 时，意识到需要使用软件来表述和检验这类猜想。

本文正是这一观察的成果。文章介绍了一个开源 SageMath软件包，旨在处理各种情形下的超几何函数：在有理数域上、在有限域上以及在 p-进域上。为实现这些功能，有必要开发新的算法。相关算法大多发表于 [4]。

目前，我们的程序包已提交，计划纳入 SageMath [3] 的未来版本。本文将简要介绍该程序包的主要功能。我们还提供了一个交互式工作表，其中汇集了下文中的所有示例，网址如下：

https://xavier.caruso.ovh/notebook/hypergeometric-functions/此交互式工作表支持编辑示例并探索新的情形。

有关该程序包的完整文档，请参阅*SageMath* 参考手册中的“任意环上的超几何函数”一节，手册可在线查阅此处。

2 设置

在 SageMath 中，超几何函数以符号环元素的形式实现。

```
输入： f =hypergeometric([1/3,2/3], [1/2], x)输入： f输出:
hypergeometric((1/3,2/3),(1/2,),x)输入:
f.parent()输出： 符号环
```

我们提出了一种实现方式，用于处理其代数和算术性质：通过创建超几何函数来访问这些性质，其中自变量是多项式环或幂级数环（的分式域）的元素。

```
输入： S.<x> =QQ[]输入： f =hypergeometric([1/3,2/3], [1/2],x)输入： f输出:
hypergeometric((1/3,2/3),(1/2,),x)输入： f.parent()输出： 定义在有理数域上的 x 中的超几何函数
```

我们强调，我们的程序包支持相当一般的参数，甚至允许下参数为非负整数，只要有适当的上参数与之匹配即可。请看：

```
输入： hypergeometric([-1], [-2],x)输出:
hypergeometric((-1,),(-2,),x)输入:
hypergeometric([-2], [-1],x)输出:
ValueError: 参数 ((-2,), (-1,)) 未定义超几何函数
```

在以下演示中，我们将始终使用的主要函数是 $f(x)$，这是一个尚不知是否为对角线的超几何函数，出自Christol的例子；$g(x)$，其中 $g(16x^2)$ 是Gessel游走的生成函数；以及较为晦涩的函数 $h(x)$，它展示了许多现象。

第二位署名作者获得了奥地利科学院维也纳大学颁发的 DOC奖学金（27150）资助。此外，他感谢法国—奥地利项目 EAGLES（ANR-22-CE91-0007 和 FWF 拨款 10.55776/I6130）提供的经费支持。作者感谢奥地利教育与国际化署（OeAD）和法国校园署通过 WTZ合作项目/Amadeus项目 FR02/2024 为研究访问提供资助。

输入：f =超几何函数([1/9,4/9,5/9], [1/3, 1], x)输入：g =超几何函数([1/2,5/6, 1], [5/3, 2],x)输入： h =超几何函数([1/5,1/5,1/5,1/5],[1/3, 3^10/5 - 1], x)

3 超几何函数 over $\mathbb{Q}$

全局有界性。 全局有界的超几何函数已由 Christol [7] 完全分类。我们在方法 is_globally_bounded() 中实现了他的判据。

输入： f.is_globally_bounded()输出： True输入： h.is_globally_bounded()输出： False

Christol 的优雅判据本质上取决于 $\{\Delta\alpha_i\}$ 和 $\{\Delta\beta_i\}$ 的小数部分的相对位置，其中 $\Delta \in \mathbb{Z}/d\mathbb{Z}$， d 是这些参数的最小公分母。

代数性。 类似地，所有代数超几何函数都已得到分类。我们在方法 is_algebraic() 中实现了相应的判据。

输入： f.is_algebraic()输出： False输入： g.is_algebraic()输出： True

若上参数与下参数之间没有整数差，Christol [7], 以及 Beukers 和 Heckman [1] 给出了分类判据；该判据与全局有界性的分类非常相似，取决于小数部分的相对位置。对于存在整数差的情况，第二作者和 Yurkevich [8]扩展了这一判据。

具有良好约化的素数。 若级数 $s(x) \in \mathbb{Q}[[x]]$的所有系数的分母都与 p互素，则称其在 p处具有良好约化。对于足够大的素数，这一性质仅取决于 p 模 d的同余类。方法 good_reduction_primes() 会计算这一性质，并输出一个由同余类决定的素数集合（第一作者已在 SageMath 中实现 [2]）。

输入： g.良好约化_约化_素数()输出： 除去 2 的所有素数集合：3、5、7、11、.....输入： h.良好约化_约化_素数()输出： 模 15 同余于 1、8、11 的素数集合，其中包括 17、167、677、857、.....、29327、29387，排除 23、83、113、173、.....、58913、58943：11、17、31、41、.....

与用于全局有界超几何函数分类的技术类似，作者在 [5] 和 [4, § 3.1] 中证明：对于足够大的素数，该级数是否具有非负的 p-进赋值，即能否模 p 约化，仅取决于它们模 d的同余类。我们的实现针对较小的素数 p，使用第 5 节中展示的方法 valuation()，检验其 p-进赋值是否为正。这一检验对所有小于该界的素数进行，此后正则性即得到保证；此外，对大于该界的每个同余类各取一个素数进行检验。我们指出与 Christol 的工作密切相关的其他方法已在 [5, § 3.1], 中详述，但尚未实现。

4 $\mathbb{F}_p$上的超几何函数

当超几何函数 $h(x)$ 在素数 p 处具有良好约化时，我们可以构造 $h(x) \bmod p \in \mathbb{F}_p[[x]]$，并研究其性质。例如，已知后者总是代数的 [6, 10, 5, 4]。除了这一结果之外，主要的工具还包括分段算子，它们使我们能够构造所谓的*Dwork* 关系 （将超几何函数表示为其他函数的 p 次幂的多项式线性组合），并最终找到湮灭多项式。我们软件包中实现了所有这些步骤。

分段。 对于整数 r，我们定义 r阶分段算子

$$\begin{array}{ccc} \mathbb{F}_p[[x]] & \longrightarrow & \mathbb{F}_p[[x]] \\ \sum_{k=0}^{\infty} a_k x^k & \mapsto & \sum_{k=0}^{\infty} a_{kp+r} x^k \end{array}$$

在 [4, §3.2], 中已证明，超几何函数的分段总可以写成一个单项式与另一个超几何函数的乘积。这些分段可以使用方法 section() 计算。

输入： f19 =f%19输入： f19.section(0)输出： 超几何函数((1/9, 4/9, 5/9), (1/3, 1), x)输入： f19.section(1)输出： 14* 超几何函数 ((1/9, 4/9, 5/9), (1/3, 1), x) 输入： f19.section(8)输出： 5*超几何函数((4/9, 5/9, 10/9), (1, 4/3), x)输入： f19.section(10)输出： 0

Dwork 关系。 根据前面的内容，可以将超几何函数写成其他超几何函数的 $\mathbb{F}_p[x]$ 线性组合，其中各项为 p 次幂。我们称这些关系为 *Dwork* 关系，并通过方法 dwork_relation() 实现。此处的输出是一个字典，其中超几何函数对应多项式系数：将键的 p 次幂乘以对应的值后求和，即可得到原超几何函数。

在： f19.dwork_relation()输出：

```
{hypergeometric((1/9,4/9,5/9),(1/3,1),x):8*x^2 + 14
*x  + 1,hypergeometric((4/9, 5/9, 10/9),(1, 4/3),
x):5*x^8 + 15*x^7}
```

湮灭多项式。 可以使用 annihilating_ore_polynomial() 方法计算超几何函数的一个约化所对应的湮灭多项式。该方法会输出一个关于 Frobenius 的多项式；若要将其视为通常的多项式，应将 Frob^n 替换为 X^{p^n}。

输入： f19.annihilating_ore_多项式()输出： (18*x^76 + 13*x^57 + 6*x^38 + 17*x^19 + 12)*Frob^2 +(12*x^38 + 11*x^32 + ... + 18*x^12 + 7)*Frob +x^30 + 16*x^29 + ... + 6*x^13 + x^12

一种执行此计算的算法已在 [5, §3.3] 中介绍。该算法通过迭代计算 Dwork 关系，得到有限多个超几何函数之间的方程组。

该算法已基本实现。我们提醒用户，当前实现依赖于该算法的一个简化版本，而尚未证明由此计算出的方程组只涉及有限多个超几何级数，因此计算可能无法终止。不过，我们相信也可以证明当前实现总能终止。

以这种方式得到的多项式绝不是不可约的。就其作为线性化多项式的本质而言，它们总是能被 $h(x)$ 整除。不过，一般而言，它们还会进一步分解。

模素数同余。 两组参数不同的超几何函数有可能得到模 p 同余的级数，如下方代码示例所示。

```
输入: T.<y> =GF(13)[]输入: h1 =超几何([1/12, 1/4], [1/2],y) 输入: h2 =超几何([1/12, 1/6], [1/3],y) 输入: h1.power_级数(1000) - h2.power_级数(1000) 输出: O(y^1000)
```

方法 is_equal_as_级数() 会检查这种情况何时发生。

```
输入: h1.is_等于_作为_series(h2)输出: 真
```

我们概述一种算法，用于检查该结论何时成立。我们的方法依赖于以下基本观察：两个级数模 p 同余，当且仅当它们的所有 r 阶截段（其中 $0 \le r < p$）都模 p同余。此外，在我们所关注的情形中，可以计算超几何级数的截段；我们已经看到，这些截段本身是超几何函数的常数倍。于是，我们可以递归地进行处理。重用用于湮灭_多项式()的论证，我们得出结论：在处理过程中只会遇到有限多个超几何级数。

为实现这一策略，我们使用两个辅助集合来记录算法的进展，即

- 集合 Q（“queued”的缩写），其中包含形如（$f_1(x)$ $f_2(x)$）的有序对；这些有序对模 p 的同余关系仍需检查，
- 集合 C（“checked”的缩写），其中包含形如（$f_1(x)$ $f_2(x)$）的有序对；这些有序对模 p 的同余关系已经检查过。

乍看之下，我们只是将问题转化为判断有限多个超几何函数对是否具有相同的约化。然而，通过检查一对此类函数的所有截面的常数项是否相同，我们可以确认这对函数在 x^{p-1}阶精度内相等。因此，只需检查常数项，即可判断它们是否相等：如果遇到的任意一对函数的常数项不相同，那么这两个超几何函数的约化就不同；否则，我们可以迭代地得出结论，它们在任意精度阶下都相等，即它们的约化相同。

算法 1： 是_同余的

输入： $h_1(x)$ $h_2(x)$ p
输出： 是否 $h_1(x) \equiv h_2(x)$（模 p）

```
Q ← {(h1(x) h2(x))};
C ← ∅;
当 Q ≠ ∅ 时执行
    弹出 f1(x) f2(x)这一对，来源为 Q;
    如果 (f1(x) f2(x)) ∈ C 或 (f2(x) f1(x)) ∈ C 则
        继续;
    对 r ← 0 到 p−1 执行
        对于 i = 1, 2，写入 Λr(fi(x))模 p，作为 ai x^ei gi(x)
          其中 ai ∈ F_p、 ei ∈ ℕ 和 gi(x) 是超几何;
        如果 a1 x^e1 ≠ a2 x^e2 属于 Fp[x]，则
            返回 false;
        如果 a1 x^e1 ≠ 0 属于 Fp[x]，则
            将 (g1(x) g2(x)) 追加到 Q;
    将 (f1(x) f2(x)) 追加到 C;
返回true;
```

p-曲率。 对于满足 $m = n - 1$的超几何函数，我们实现了相关超几何微分算子 $\mathrm{L}(\boldsymbol{\alpha}, \boldsymbol{\beta}; x)$ 的 p-曲率，也就是作用于 $\mathbb{F}_p[x]\partial/\mathrm{L}(\boldsymbol{\alpha}, \boldsymbol{\beta}; x)\mathbb{F}_p[x]\partial$ 上的线性映射 ∂^p 的矩阵表示。

可通过方法 p_curvature() 访问。其余秩决定了微分算子 $\mathrm{L}(\boldsymbol{\alpha}, \boldsymbol{\beta}; x)$ 在 $\mathbb{F}_p(x)$ 中的 $\mathbb{F}_p(x^p)$ 维解空间。

给定超几何函数的 p-曲率余秩会在 p 中均匀变化，[5, 命题 3.1。20] 对于足够大的素数 p。

相应的同余类和例外情况通过定义在 $\mathbb{Q}$上的超几何函数的 methodp_curvature_coranks() 方法实现。

```
输入: f5 =f%5输入: f5.p_曲率()输出:
[            0 2/(x^5 + 4*x^4)1/(x^4 + 4*x^3)]
[            0             0             0]
[            0             0             0]输入:
f.p_曲率 _ 余秩 () 输出: {1: 空的素数集合 , 2: 排除 3 的所有素数集合: 2, 5, 7, 11, ..., 3: 空的素数集合}
```

5 $\mathbb{Q}_p$ 上的超几何函数

最后，我们讨论定义在 p 进数域 $\mathbb{Q}_p$ 上、参数为有理数的超几何函数。对于 $x \in \mathbb{Q}_p$，我们用 $\mathrm{val}_p(x)$ 表示其 p 进赋值，并用 $\|x\|_p = p^{-\mathrm{val}_p(x)}$ 表示其 p 进范数。

收敛半径。 与复数情形类似，级数 $s(x)$ 的 p-adic 收敛半径定义为 $= \sum a_k x^k \in \mathbb{Q}_p[[x]]$

$$\liminf_{k\to\infty} \|a_k\|_p^{-1/k}.$$

当 $\|a\|_p$（其中 $a \in \mathbb{Q}_p$）小于这个临界值时，级数 $s(a)$ 在 $\mathbb{Q}_p$ 中收敛。方法 log_radius_of_convergence() 计算超几何级数的 p-adic 收敛半径以 p 为底的对数。

```
输入: hp5 =h.change_ring(Qp(5))输入: hp5.log_radius_of_convergence()输出: -7/2
```

计算 ${}_nF_m(\boldsymbol{\alpha}, \boldsymbol{\beta}; x)$ 的对数 p-进收敛半径的算法，基本遵循 [4, §2] 中的讨论。首先，我们将 $\boldsymbol{\alpha} = \boldsymbol{\alpha}' \sqcup \boldsymbol{\alpha}'', \boldsymbol{\beta} = \boldsymbol{\beta}' \sqcup \boldsymbol{\beta}''$ 分成若干部分，其中 $\boldsymbol{\alpha}', \boldsymbol{\beta}'$ 恰好包含

参数中的 p-进整数。然后，假设 ${}_nF_m(\boldsymbol{\alpha}, \boldsymbol{\beta}; x)$ 不是多项式，其对数收敛半径由以下显式公式给出

$$\frac{n'-m'-1}{p-1} + \sum_{\alpha\in\boldsymbol{\alpha}''} \mathrm{val}_p(\alpha) - \sum_{\beta\in\boldsymbol{\beta}''} \mathrm{val}_p(\beta),$$

其中 n' 和 m' 分别是 $\boldsymbol{\alpha}'$ 和 $\boldsymbol{\beta}'$ 的基数。相反，当 ${}_nF_m(\boldsymbol{\alpha}, \boldsymbol{\beta}; x)$ 是多项式时，其对数收敛半径自然是无穷大。

赋值。对于 $v \in \mathbb{Q}$，我们称

$$\mathrm{val}_{p,v}(s(x)) := \min_{k\geq 0} \mathrm{val}_p(a_k) + vk$$

级数 $s(x)$ 的 v-漂移 p-进赋值。对于 $v = 0$，它显然与 p-进 Gauss 赋值一致，该赋值作用于 $s(x)$；对于任意的 v，它可以解释为 $s(x)$ 在以 0 为中心、p-进半径为 p^v 的圆盘上的 p-进赋值。特别地，当 v 大于对数 p-进收敛半径时，它为 $-\infty$。

方法 valuation() 计算超几何函数的 p-进赋值；传入参数 v 时，则计算 v-漂移的 p-进赋值。此外，还可以传入选项 position=True，以同时输出系数在 x^k 中达到该赋值的最小指标 k。

```
输入： fp5 =f.change_
ring(Qp(5))输入： fp5.valuation()
输出： 0输入： hp3 =h.change_
ring(Qp(3))输入：
hp3.valuation(position=True)输出：
(-4,2) 输入： hp5.valuation() 输出：
-Infinity 输入：
hp5.valuation(-7/2)输出： 0
```

作者在 [4,§2.2] 中介绍了一种算法，用于计算超几何级数的 v-漂移 p-adic 赋值。该算法依赖于热带半环上的递归，以及 Floyd-Warshall 算法来计算热带矩阵的弱传递闭包。正如 [4, 备注 2.6]中所解释的，记录取得系数 h_k 的赋值时所对应的最小指标 k，很容易实现。

p-adic 求值。当 a 是一个范数小于收敛半径的 p-adic 数时，值 $h(a)$是有意义的。本软件包允许使用以下显然的语法来计算它。

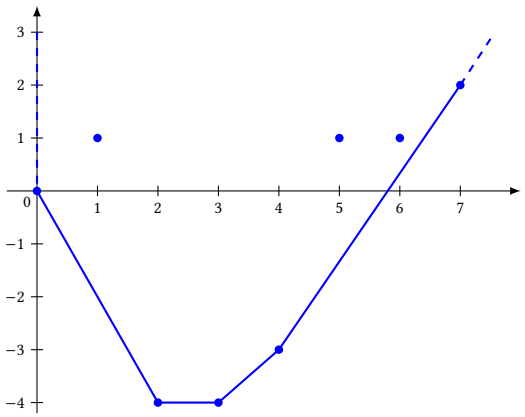
```
输入： fp5(5)输出： 1 + 3*5^2 + 5^4 +... +
O(5^20)输入： hp3(1/3)输出： 3^-5 + 2*3^-
1 + 1 + 2*3 +... +O(3^13)
```

所实现的算法概述见 [4, §2.4]：给定一个 p-进数 $a \in \mathbb{Q}_p$，其 $\mathrm{val}_p(a) = v$ 位于收敛半径内，并给定精度 N，我们首先计算一个界 K，使得对所有 $k > K$，$\mathrm{val}_p(h_k a^k) > N$。因此 $h(a) = \sum_{k=0}^{K} h_k a^k + O(p^N)$。$K$ 的计算取决于一个参数的选取，该参数介于 v和收敛半径之间；我们选择该参数的启发式方法也在上述文献中作了解释。

牛顿多边形。级数 $\sum_k a_k x^k$的牛顿多边形是 $\mathbb{R}^2$ 中由点 (k, v) 构成的凸包，其中 $v \geq \mathrm{val}_p(h_k)$。

对于超几何级数，可以使用 newton_polygon() 方法计算其牛顿多边形。传入参数 v，且令其小于对数 p-adic 收敛半径，有助于处理牛顿多边形具有无限多个斜率的情况。这样会缩小超几何级数的定义域，从而得到牛顿多边形的近似。

```
输入： hp3.newton_polygon()输出： ValueError：无限牛顿
多边形；请通过指定小于 2 的对数半径来截断它输入： NP =
hp3.newton_polygon(7/4)输入： NP输出： 无限牛顿多边形，有
5 个顶点：(0, 0)、(2, -4)、(3, -4)、(4, -3)、(7, 2)，并以斜率为
7/4 的无限直线结束输入： NP.plot() +point([(i, hp3[i].
valuation()) for i in range(8)])
```



我们实现了 [4, §2.3]，中的算法，它基本上是用于计算漂移赋值的算法的一种推广。

参考文献

[1] F. Beukers 和 G. Heckman。1989。超几何函数 ${}_nF_{n-1}$ 的单值化。《数学发明》，95，2，325–354。doi:10.1007/BF01393900。[2] X. Caruso。2025。由同余条件定义的素数子集。https://g ith.com/sagemath/sage/pull/41122。[3] X. Caruso 和 F. Fürnsinn。2025。超几何函数的代数与模性质。https://github.com/sagemath/sage/pull/41113。[4] X. Caruso 和 F. Fürnsinn。2026。超几何函数代数与算术属性的算法。预印本：arXiv:2601.16105 (math.NT)。[5] X. Caruso、F. Fürnsinn 和 D. Vargas-Montoya。2025。D有限级数模 p 约化的伽罗瓦群。预印本：arXiv:2504.09429 (math.NT)。[6] G. Christol。1986。函数与代数元素。《太平洋数学杂志》，125，1，1–37。doi:10.2140/pjm.1986.125.1。[7] G. Christol。1986。有界超几何函数。超度量分析工作组。第8号报告 14，1–16。http://www.numdam.org/item /GAU_1986-1987__14__A4_0.pdf。[8] F. Fürnsinn 和 S. Yurkevich。2024。具有任意参数的超几何函数的代数性。《伦敦数学会会刊》，blms.13103。doi:10.1112/blms.13103。[9] D. Vargas-Montoya。2024。广义超几何级数模 p 的代数性 ${}_nF_{n-1}$ 259, 273 321 10 1016 202。《数论杂志》，4 01 004，–。doi: . /j.jnt. . . .[10] D. Vargas-Montoya。2021。模 p的代数性、超几何级数与强弗贝尼乌斯结构。《法国数学会通报》。doi:10.24033/bsmf.2834。