# Advanced Mathematics

Maple 2022 includes a very large number of improvements that strengthen the math engine, expanding Maple's abilities to handle new areas of mathematics and solve harder problems, faster.  In addition to the mathematical improvements described in more detail elsewhere, Maple 2022 enhances support for many areas of mathematics, and improves fundamental routines that are used regularly both by users and by other Maple commands.

## Improvements to Core Mathematics Routines

### Integration

For Maple 2022, we put a large focus on improving integration by correcting known weaknesses, and, as a result, Maple can solve more integration problems than ever.  The following are some examples.

int is now smarter about changes of variables that can avoid integrating term by term:

```
> int(x*(x^2+1)^99,x);
```

$$\frac{\left(x^2+1\right)^{100}}{200}$$

Significant improvements to elliptic integration led to these new results:

```
> VectorCalculus:-ArcLength(z-> <sqrt(-z+2), sqrt(z-1), z>, 1 .. 2)
  ;
```

$$\sqrt{2}\ \mathrm{EllipticE}\left(\frac{\sqrt{2}}{2}\right)$$

```
> int(sqrt((-5*x^2 + 81)/(-9*x^2 + 81)), x = 0 .. 3);
```

$$3\ \mathrm{EllipticE}\left(\frac{\sqrt{5}}{3}\right)$$

```
> int((1 - I*sqrt(1 - z^3))^4, z=-1..1);
```

$$\frac{64\,\mathrm{I}\sqrt{2}}{55} - \frac{12\,\mathrm{I}\,3^{1/4}\sqrt{2}\,\mathrm{EllipticF}\left(\frac{\mathrm{I}}{2} + \frac{\sqrt{3}}{2}, \frac{\sqrt{3}}{2} - \frac{\mathrm{I}}{2}\right)}{55} - \frac{54}{7}$$
$$+ \frac{4\,\mathrm{I}\,3^{3/4}\sqrt{2}\,\mathrm{EllipticF}\left(\frac{\mathrm{I}}{2} + \frac{\sqrt{3}}{2}, \frac{\sqrt{3}}{2} - \frac{\mathrm{I}}{2}\right)}{55}$$
$$+ \frac{12\sqrt{2}\sqrt{3-\mathrm{I}\sqrt{3}}\,\mathrm{EllipticF}\left(\frac{\sqrt{2}\sqrt{3-\mathrm{I}\sqrt{3}}\sqrt{3}}{6}, \frac{\sqrt{3}}{2} - \frac{\mathrm{I}}{2}\right)}{55}$$
$$+ \frac{4\,\mathrm{I}\sqrt{2}\sqrt{3}\sqrt{3-\mathrm{I}\sqrt{3}}\,\mathrm{EllipticF}\left(\frac{\sqrt{2}\sqrt{3-\mathrm{I}\sqrt{3}}\sqrt{3}}{6}, \frac{\sqrt{3}}{2} - \frac{\mathrm{I}}{2}\right)}{55}$$
$$- \frac{4\,3^{3/4}\sqrt{2}\,\mathrm{EllipticF}\left(\frac{\mathrm{I}}{2} + \frac{\sqrt{3}}{2}, \frac{\sqrt{3}}{2} - \frac{\mathrm{I}}{2}\right)}{55}$$
$$- \frac{12\,3^{1/4}\sqrt{2}\,\mathrm{EllipticF}\left(\frac{\mathrm{I}}{2} + \frac{\sqrt{3}}{2}, \frac{\sqrt{3}}{2} - \frac{\mathrm{I}}{2}\right)}{55}$$

```
> int((z^3 - I*sqrt(1 - z^4))^4, z=-1..1);
```

$$\frac{42914}{45045}$$

```
> int(1/(sqrt(z-1)*sqrt(z-2)*sqrt(3-z)),z=0..1);
```

$$-\sqrt{2}\,\mathrm{EllipticK}\left(\frac{\sqrt{2}}{2}\right) + \sqrt{2}\,\mathrm{EllipticF}\left(\frac{\sqrt{2}\sqrt{3}}{3}, \frac{\sqrt{2}}{2}\right)$$

This integral gives a new result:

```
> int( sin(Pi*(x^2+y^2)), y = 0 .. (-x^2+1)^(1/2) ) assuming x>0,
  x<1;
```

$$\frac{\left(\cos\left(\pi x^2\right)\mathrm{FresnelS}\left(\sqrt{-x^2+1}\,\sqrt{2}\right) + \sin\left(\pi x^2\right)\mathrm{FresnelC}\left(\sqrt{-x^2+1}\,\sqrt{2}\right)\right)\sqrt{2}}{2}$$

Previously this gave an error:

```
> int(1-abs(x)-abs(x-y)+abs(-1+abs(x)+abs(x-y)),x) assuming -1<x,
  x<1;
```

$$x - \left( \begin{cases} -\frac{x^2}{2} & x \le 0 \\ \frac{x^2}{2} & 0 < x \end{cases} \right) - \left( \begin{cases} -\frac{1}{2}x^2 + yx & x \le y \\ \frac{1}{2}x^2 - yx + y^2 & y < x \end{cases} \right)$$

$$+ \left( \begin{cases} \begin{cases} \begin{cases} -x^2 + yx - x & x \le -\frac{1}{2} + \frac{y}{2} \\ x^2 - yx + x + \frac{1}{2}y^2 - y + \frac{1}{2} & x \le 0 \\ -yx + x + \frac{1}{2}y^2 - y + \frac{1}{2} & x \le y \\ -x^2 + yx + x - \frac{1}{2}y^2 + \frac{1}{2} - y & x \le \frac{1}{2} + \frac{y}{2} \\ x^2 - yx - x + 1 & \frac{1}{2} + \frac{y}{2} < x \end{cases} & y < 1 \\ \begin{cases} \begin{cases} -x^2 & x \le 0 \\ 0 & 0 < x \end{cases} & -1 + y = 0 \\ \begin{cases} -x^2 + yx - x & x \le 0 \\ (-1 + y)\,x & 0 < x \end{cases} & \mathit{otherwise} \end{cases} & \mathit{otherwise} \end{cases} & 0 < y \\ \begin{cases} \begin{cases} -x^2 + yx - x & x \le -\frac{1}{2} + \frac{y}{2} \\ x^2 - yx + x + \frac{1}{2}y^2 - y + \frac{1}{2} & x \le y \\ yx + x - \frac{1}{2}y^2 + \frac{1}{2} - y & x \le 0 \\ -x^2 + yx + x - \frac{1}{2}y^2 + \frac{1}{2} - y & x \le \frac{1}{2} + \frac{y}{2} \\ x^2 - yx - x + 1 & \frac{1}{2} + \frac{y}{2} < x \end{cases} & -y < 1 \\ \begin{cases} \begin{cases} 0 & x \le 0 \\ x^2 & 0 < x \end{cases} & -1 - y = 0 \\ \begin{cases} -(1 + y)\,x & x \le 0 \\ x^2 - yx - x & 0 < x \end{cases} & \mathit{otherwise} \end{cases} & \mathit{otherwise} \end{cases} & \mathit{otherwise} \end{cases} \right)$$

This was previously unevaluated:

```
> int(1-abs(x)-abs(x-y)+abs(1-abs(x)-abs(x-y)),[x = -1 .. 1, y = -1
  .. 1]);
```

$$\frac{4}{3}$$

This integral gives a new result:

```
> int(z^(1/5)/(z^2+1),z = -infinity .. infinity, method=FTOC);
```

$$-\frac{\pi\left(-\sqrt{2}\,\sqrt{5}\,\sqrt{5-\sqrt{5}}\,+2\,\mathrm{I}\sqrt{5}\,-5\sqrt{2}\,\sqrt{5-\sqrt{5}}\,-10\,\mathrm{I}\right)}{2\sqrt{10-2\sqrt{5}}\,\sqrt{10+2\sqrt{5}}}$$

This was previously unevaluated:

```
> int(((z-1)/(z+1)^(1/2))^(2*x+1), z=0..1);
```

$$-\frac{1}{\Gamma\left(x+\frac{1}{2}\right)(-1+2x)}\left((-1)^{2x}2^{x+\frac{1}{2}}\left(2^{\frac{3}{2}+x}\mathrm{hypergeom}\left(\left[-1-2x,\frac{1}{2}-x\right],\left[\frac{3}{2}-x\right],\frac{1}{2}\right)\Gamma\left(x+\frac{1}{2}\right)+4\,\Gamma\left(-\frac{3}{2}-x\right)\Gamma(2x+2)\,x-2\,\Gamma\left(-\frac{3}{2}-x\right)\Gamma(2x+2)\right)\right)$$

This was previously unevaluated:

```
> int(Sum((-1)^x*((99/100*Pi*z-99/100*Pi)/(99/100*Pi*z+1/100*Pi)^
  (1/2))^(2*x+1)/(2*x+1)!,x = 0 .. infinity), z=0..1);
```

$$-\frac{1}{99}\left(\sqrt{\pi}\left(\sum_{x=0}^{\infty}\frac{1}{(2x+1)!\,\Gamma\left(x+\frac{1}{2}\right)(-1+2x)}\left((-1)^{3x}\pi^{x}\left(2^{2x+2}5^{2x+1}\mathrm{hypergeom}\left(\left[-1-2x,\frac{1}{2}-x\right],\left[\frac{3}{2}-x\right],\frac{1}{100}\right)\Gamma\left(x+\frac{1}{2}\right)+200\,\Gamma\left(-\frac{3}{2}-x\right)\Gamma(2x+2)\,x-100\,\Gamma\left(-\frac{3}{2}-x\right)\Gamma(2x+2)\right)\right)\right)\right)$$

This was previously unevaluated:

```
> int(int(f(y), y=a..b), y=a..b);
```

$$(b-a)\left(\int_{a}^{b}f(y)\,\mathrm{d}y\right)$$

This was previously unevaluated:

```
> int(((1/t)^(1 - 2*n)  - 1) / (t - 1), t=0..x);
```

$$-\frac{1}{2\sin(2\pi n)\,n}\left(2\pi\left(\frac{1}{x}\right)^{-2n}(-x)^{-2n}n+2\left(\frac{1}{x}\right)^{-2n}\mathrm{LerchPhi}\left(\frac{1}{x},1,-2n\right)n\sin(2\pi n)+\left(\frac{1}{x}\right)^{-2n}\sin(2\pi n)+2n\ln(-x)\sin(2\pi n)+2n\ln\left(\frac{x-1}{x}\right)\sin(2\pi n)\right)$$

This was previously unevaluated:

```
> int(exp(t^3),t=0..x);
```

$$-\frac{\left(-x^3\right)^{2/3}\left(2\pi\sqrt{3}-3\,\Gamma\left(\frac{1}{3},-x^3\right)\Gamma\left(\frac{2}{3}\right)\right)}{9x^2\,\Gamma\left(\frac{2}{3}\right)}$$

This was previously unevaluated:

```
> int(cos(x)^(1/3)*sin(x)^2,x=-Pi/2..Pi/2);
```

$$\frac{9\,\Gamma\left(\frac{2}{3}\right)\Gamma\left(\frac{5}{6}\right)}{7\sqrt{\pi}}$$

This integral gives a new result:

```
> int(arctan(z)*(z-1)^(1/3), z=0..1);
```

$$-\frac{1}{\left(\mathrm{I}\sqrt{3}+1\right)^2\left(\mathrm{I}\sqrt{3}-1\right)^2\left(-1+\mathrm{I}\right)^{2/3}}\left(6\left(-3\,\mathrm{I}\left(-1+\mathrm{I}\right)^{2/3}\sqrt{3}+2\left(-1+\mathrm{I}\right)^{2/3}\left(\sum_{\_R=RootOf\left(\_Z^6+2\,\_Z^3+2\right)}\frac{\ln\left(\mathrm{I}\sqrt{3}-2\,\_R+1\right)}{\_R^2}\right)-2\left(-1+\mathrm{I}\right)^{2/3}\left(\sum_{\_R=RootOf\left(\_Z^6+2\,\_Z^3+2\right)}\frac{\ln\left(-\_R\right)}{\_R^2}\right)-3\left(-1+\mathrm{I}\right)^{2/3}-4\,\mathrm{I}\,\pi\right)\right)$$

Better handling of Dirac in int led to this improved result:

```
> int(1/exp(I*q*a), q = -infinity .. infinity);
```

$$2\pi\,\mathrm{Dirac}(a)$$

This new result is due to improved discontinuity handling in the presence of Heaviside:

```
> int(Heaviside(-y^2+B)/(-y^2+B)^(1/2), y=-infinity .. infinity)
  assuming B>0;
```

$$\pi$$

This new result is due to an improvement in simplification of logarithms:

```
> int(cos(x)^3*sin(x)^n,x);
```

$$\frac{1}{8\,(n+3)}\Biggl(\mathrm{I}\,\bigl(\mathrm{e}^{\mathrm{I}\,x}\bigr)^{-n}\,\bigl(\mathrm{e}^{2\,\mathrm{I}\,x}$$

$$-\,1\bigr)$$

$$^{n}\left(\frac{1}{2}\right)^{n}$$

$$\mathrm{e}^{\frac{\mathrm{I}}{2}\,\bigl(\pi\,n\,\mathrm{csgn}\bigl(\mathrm{I}\,\bigl(\mathrm{e}^{2\,\mathrm{I}\,x}-1\bigr)\bigr)\,\mathrm{csgn}\bigl(\mathrm{I}\,\mathrm{e}^{-\mathrm{I}\,x}\bigr)\,\mathrm{csgn}(\sin(x))+\pi\,n\,\mathrm{csgn}\bigl(\mathrm{I}\,\bigl(\mathrm{e}^{2\,\mathrm{I}\,x}-1\bigr)\bigr)\,\mathrm{csgn}(\sin(x))^{2}}$$

$$+\,\pi\,n\,\mathrm{csgn}\bigl(\mathrm{I}\,\mathrm{e}^{-\mathrm{I}\,x}\bigr)\,\mathrm{csgn}(\sin(x))^{2}+\pi\,n\,\mathrm{csgn}(\sin(x))^{3}-\pi\,n\,\mathrm{csgn}(\sin(x))\,\mathrm{csgn}(\mathrm{I}\sin(x))^{2}-\pi\,n\,\mathrm{csgn}(\mathrm{I}\sin(x))^{3}$$

$$+\,\pi\,n\,\mathrm{csgn}(\sin(x))\,\mathrm{csgn}(\mathrm{I}\sin(x))+\pi\,n\,\mathrm{csgn}(\mathrm{I}\sin(x))^{2}-\pi\,n-6\,x\bigr)\Biggr)$$

$$+\,\frac{1}{8\,(n+3)\,(1+n)}\Biggl(\mathrm{I}\left(\frac{1}{2}\right)^{n}\bigl(\mathrm{e}^{2\,\mathrm{I}\,x}-1\bigr)^{n}\,\bigl(\mathrm{e}^{\mathrm{I}\,x}\bigr)^{-n}\,(n$$

$$+\,9)$$

$$\mathrm{e}^{\frac{\mathrm{I}}{2}\,\bigl(\pi\,n\,\mathrm{csgn}\bigl(\mathrm{I}\,\bigl(\mathrm{e}^{2\,\mathrm{I}\,x}-1\bigr)\bigr)\,\mathrm{csgn}\bigl(\mathrm{I}\,\mathrm{e}^{-\mathrm{I}\,x}\bigr)\,\mathrm{csgn}(\sin(x))+\pi\,n\,\mathrm{csgn}\bigl(\mathrm{I}\,\bigl(\mathrm{e}^{2\,\mathrm{I}\,x}-1\bigr)\bigr)\,\mathrm{csgn}(\sin(x))^{2}}$$

$$+\,\pi\,n\,\mathrm{csgn}\bigl(\mathrm{I}\,\mathrm{e}^{-\mathrm{I}\,x}\bigr)\,\mathrm{csgn}(\sin(x))^{2}+\pi\,n\,\mathrm{csgn}(\sin(x))^{3}-\pi\,n\,\mathrm{csgn}(\sin(x))\,\mathrm{csgn}(\mathrm{I}\sin(x))^{2}-\pi\,n\,\mathrm{csgn}(\mathrm{I}\sin(x))^{3}$$

$$+\,\pi\,n\,\mathrm{csgn}(\sin(x))\,\mathrm{csgn}(\mathrm{I}\sin(x))+\pi\,n\,\mathrm{csgn}(\mathrm{I}\sin(x))^{2}-\pi\,n-2\,x\bigr)\Biggr)$$

$$-\,\frac{1}{8\,(n+3)}\Biggl(\mathrm{I}\left(\frac{1}{2}\right)^{n}\bigl(\mathrm{e}^{2\,\mathrm{I}\,x}$$

$$-\,1\bigr)$$

$$^{n}\bigl(\mathrm{e}^{\mathrm{I}\,x}\bigr)^{-n}$$

$$\mathrm{e}^{\frac{\mathrm{I}}{2}\,\bigl(\pi\,n\,\mathrm{csgn}\bigl(\mathrm{I}\,\bigl(\mathrm{e}^{2\,\mathrm{I}\,x}-1\bigr)\bigr)\,\mathrm{csgn}\bigl(\mathrm{I}\,\mathrm{e}^{-\mathrm{I}\,x}\bigr)\,\mathrm{csgn}(\sin(x))+\pi\,n\,\mathrm{csgn}\bigl(\mathrm{I}\,\bigl(\mathrm{e}^{2\,\mathrm{I}\,x}-1\bigr)\bigr)\,\mathrm{csgn}(\sin(x))^{2}}$$

$$+\,\pi\,n\,\mathrm{csgn}\bigl(\mathrm{I}\,\mathrm{e}^{-\mathrm{I}\,x}\bigr)\,\mathrm{csgn}(\sin(x))^{2}+\pi\,n\,\mathrm{csgn}(\sin(x))^{3}-\pi\,n\,\mathrm{csgn}(\sin(x))\,\mathrm{csgn}(\mathrm{I}\sin(x))^{2}-\pi\,n\,\mathrm{csgn}(\mathrm{I}\sin(x))^{3}$$

$$+ \pi n \operatorname{csgn}(\sin(x)) \operatorname{csgn}(\mathrm{I} \sin(x)) + \pi n \operatorname{csgn}(\mathrm{I} \sin(x))^{2} - \pi n + 6 x\Big)\Big)$$

$$- \frac{1}{8 (n+3) (1+n)} \Bigg(\mathrm{I} \left(\frac{1}{2}\right)^{n} \left(\mathrm{e}^{2 \mathrm{I} x} - 1\right)^{n} \left(\mathrm{e}^{\mathrm{I} x}\right)^{-n} (n$$

$$+ 9)$$

$$\mathrm{e}^{\frac{\mathrm{I}}{2}} \left(\pi n \operatorname{csgn}\left(\mathrm{I} \left(\mathrm{e}^{2 \mathrm{I} x} - 1\right)\right) \operatorname{csgn}\left(\mathrm{I}\, \mathrm{e}^{-\mathrm{I} x}\right) \operatorname{csgn}(\sin(x)) + \pi n \operatorname{csgn}\left(\mathrm{I} \left(\mathrm{e}^{2 \mathrm{I} x} - 1\right)\right) \operatorname{csgn}(\sin(x))^{2}\right.$$

$$+ \pi n \operatorname{csgn}\left(\mathrm{I}\, \mathrm{e}^{-\mathrm{I} x}\right) \operatorname{csgn}(\sin(x))^{2} + \pi n \operatorname{csgn}(\sin(x))^{3} - \pi n \operatorname{csgn}(\sin(x)) \operatorname{csgn}(\mathrm{I} \sin(x))^{2} - \pi n \operatorname{csgn}(\mathrm{I} \sin(x))^{3}$$

$$+ \pi n \operatorname{csgn}(\sin(x)) \operatorname{csgn}(\mathrm{I} \sin(x)) + \pi n \operatorname{csgn}(\mathrm{I} \sin(x))^{2} - \pi n + 2 x\Big)\Big)$$

There is a new result from inttrans:-fourier:

```
> inttrans:-fourier(x^n*ln(x),x,s) assuming n::posint;
```

$$ft := \mathrm{I}^{n} \pi \left(\mathrm{I} \pi \operatorname{Dirac}(n, s) - 2 \left(\sum_{\_k1=0}^{n} \binom{n}{\_k1} \left(\frac{\mathrm{d}^{\_k1}}{\mathrm{d}s^{\_k1}} \operatorname{Heaviside}(s)\right) \operatorname{pochhammer}(-n + \_k1, n\right.\right.$$

$$\left.\left. - \_k1)\, s^{-1-n+\_k1}\right)\right)$$

Some improvements to IntegrationTools:-Change led to these new results:

```
> IntegrationTools:-Change(Int((-3*x^2+x+2*Pi)*x, x=0..1), u = -3*
  x^2+x+2*Pi);
```

$$- \frac{\displaystyle\left(\int_{2\pi}^{\frac{1}{12}+2\pi} \frac{\left(24\pi - 12u + 1 - \sqrt{1 - 12u + 24\pi}\right) u}{1 - 12u + 24\pi} \,\mathrm{d}u\right)}{6}$$

$$+ \frac{\displaystyle\left(\int_{-2+2\pi}^{\frac{1}{12}+2\pi} \frac{\left(24\pi - 12u + 1 + \sqrt{1 - 12u + 24\pi}\right) u}{1 - 12u + 24\pi} \,\mathrm{d}u\right)}{6}$$

## Differential Equations

Improved solvers for partial differential equations enable Maple 2022 to solve problems it couldn't solve before. For example, pdsolve can now handle this problem:

```
> pde[N3] := diff(u(r, theta, t), t, t) = (diff(u(r, theta, t), r)+
  r*(diff(u(r, theta, t), r, r))+(diff(u(r, theta, t), theta,
  theta))/r)/(4*r);
  iv[N3] := u(1, theta, t) = 0, (D[2](u))(r, 0, t) = 0, (D[2](u))
  (r, Pi, t) = 0, u(r, theta, 0) = r^2*cos(2*theta), (D[3](u))(r,
  theta, 0) = 0;
  pdsolve([pde[N3], iv[N3]], u(r, theta, t), HINT = boundedseries(r
  = [0])) assuming r >= 0, r <= 1, theta >= 0, theta <= Pi;
```

$$pde_{N3} := \frac{\partial^2}{\partial t^2}\, u(r,\theta,t) = \frac{\frac{\partial}{\partial r}\, u(r,\theta,t) + r\left(\frac{\partial^2}{\partial r^2}\, u(r,\theta,t)\right) + \frac{\frac{\partial^2}{\partial \theta^2}\, u(r,\theta,t)}{r}}{4\, r}$$

$$iv_{N3} := u(1,\theta,t) = 0, \mathrm{D}_2(u)(r,0,t) = 0, \mathrm{D}_2(u)(r,\pi,t) = 0, u(r,\theta,0) = r^2\cos(2\,\theta), \mathrm{D}_3(u)(r,\theta,0) = 0$$

$$u(r,\theta,t) = \sum_{n=1}^{\infty}\left(\_C9_{0,n}\, J_0(\lambda_n\, r)\cos\left(\frac{\lambda_n\, t}{2}\right) + \frac{1}{J_1(\mathrm{BesselJZeros}(2,n))^2\,\mathrm{BesselJZeros}(2,n)^3}\left(\left(-2\,\mathrm{BesselJZeros}(2,n)^2\, J_1(\mathrm{BesselJZeros}(2,n)) - 8\,\mathrm{BesselJZeros}(2,n)\, J_0(\mathrm{BesselJZeros}(2,n)) + 16\, J_1(\mathrm{BesselJZeros}(2,n))\right) J_2(\mathrm{BesselJZeros}(2,n)\, r)\cos(2\,\theta)\cos\left(\frac{\lambda_n\, t}{2}\right)\right)\right)\ \textbf{where}\ \left\{\lambda_n = \mathrm{BesselJZeros}(0,n) \wedge 0 \le \lambda_n\right\}$$

The FindODE command in the DEtools package has been improved. It has a few new options, and it is more efficient on many examples. The following examples took more than twice as long in earlier versions of Maple:

```
> with(DEtools):
```

```
> DE := CodeTools:-Usage(FindODE(arctan(z)^2+sin(z)^4+log(1+z)^5, y
  (z), 12)):
memory used=86.33MiB, alloc change=32.00MiB, cpu time=870.00ms, real
time=835.00ms, gc time=150.50ms
```

```
> PDEtools:-difforder(DE,z);
```

$$11$$

```
> DE := CodeTools:-Usage(FindODE((3*z+5*z^7+11*z^13)*log(1+z^3+z^7)
  +arctanh(z)*cos(z)^5, y(z), 15)):
```

```
memory used=1.01GiB, alloc change=50.23MiB, cpu time=8.86s, real
time=8.23s, gc time=1.20s
```

```
> PDEtools:-difforder(DE,z);
```

$$14$$

## Integral Equations

The intsolve command now has an option to specify if an approximate solution using collocation is to be determined numerically or symbolically. For example:

```
> ie := f(x) + int( f(y) * piecewise( x < y, x * ( 5 - y ), y * ( 5
  - x ) ), y = 0 .. 5 ) - x = 0;
```

$$ie := f(x) + \int_0^5 f(y) \left( \begin{cases} x\,(5-y) & x < y \\ y\,(5-x) & \mathit{otherwise} \end{cases} \right) \mathrm{d}y - x = 0$$

```
> intsolve( ie, f(x), 'method' = 'collocation', 'order' = 5,
  'numeric' = 'true' );
```

$$f(x) = 0.0249169435217379\,x^5 - 0.235304198570564\,x^4 + 0.791580191676118\,x^3 - 1.08998977144344\,x^2 + 0.500379185550479\,x + 8.41923579640045 \times 10^{-15}$$

```
> intsolve( ie, f(x), 'method' = 'collocation', 'order' = 5,
  'numeric' = 'false' );
```

$$f(x) = \frac{15}{602}\,x^5 - \frac{174375}{741062}\,x^4 + \frac{293305}{370531}\,x^3 - \frac{403875}{370531}\,x^2 + \frac{185406}{370531}\,x$$

## Simplification

Simplification has been improved in a number of areas.

The arctrig functions now perform some new automatic simplification by pulling out signs:

```
> arccos(csgn(z)*z),arccot(csgn(z)*z),arcsec(csgn(z)*z);
```

$$\frac{\pi}{2} - \operatorname{csgn}(z)\,\arcsin(z),\ \frac{\pi}{2} - \operatorname{csgn}(z)\,\arctan(z),\ \frac{\pi}{2} - \operatorname{csgn}(z)\,\operatorname{arccsc}(z)$$

The following inverse trigonometric identities are now recognized by simplify:

$$\operatorname{arctan}(z) + \operatorname{arccot}(z) = \operatorname{arccos}(z) + \operatorname{arcsin}(z) = \operatorname{arcsec}(z) + \operatorname{arccsc}(z) = \frac{\pi}{2}$$

```
> SA := () -> simplify(args, 'arctrig'):
```

```
> SA(arcsin(z) + arccos(z)),
  SA(arcsec(z) + arccsc(z));
```

$$\frac{\pi}{2}, \frac{\pi}{2}$$

simplify and combine now recognize the following identities:

```
> SA(arccot(1)+arccot(2)+arccot(3));
  combine(arccot(1)+arccot(2)+arccot(3));
```

$$\frac{\pi}{2}$$

$$\frac{\pi}{2}$$

simplify now recognizes when two simplifications can be combined:

```
> SA(arctan(cos(2)/sin(2)));
```

$$\frac{\pi}{2} - 2$$

When no cancellation is available, simplify converts to arctan, arcsin, or arccsc by default:

```
> SA(arccos(x) + Pi/2),
  SA(arcsec(x) + Pi/2);
```

$$\pi - \operatorname{arcsin}(x), \pi - \operatorname{arccsc}(x)$$

When one form is more compact, it uses that form instead of the default:

```
SA(Pi/2 - arctan(z)),
SA(Pi/2 - arcsin(z)),
SA(Pi/2 - arccos(z)),
SA(Pi/2 - arcsec(z)),
SA(Pi/2 - arccsc(z));
```

$$\operatorname{arccot}(z), \operatorname{arccos}(z), \operatorname{arcsin}(z), \operatorname{arccsc}(z), \operatorname{arcsec}(z)$$

Trigonometric simplification now makes use of formulae such as the following where applicable:

$$(1-\cos(2x))^{\frac{1}{2}} \quad \left(2\sin(x)^2\right)^{\frac{1}{2}}$$

```
> simplify(-1+1/2*sin(1/4*Pi+1)*(2+2*sin(2))^(1/2)-1/2*cos(1/4*
  Pi+1)*(2-2*sin(2))^(1/2));
```

$$0$$

Improved simplification of nested integrals:

```
> simplify(Int(((x-1)*(-Heaviside(y)*Heaviside(x)*Heaviside(y-x)+
  Heaviside(y)*Heaviside(x))+x*Heaviside(y)*Heaviside(x)*Heaviside
  (y-x))*((x-1)*(-Heaviside(y)*Heaviside(x)*Heaviside(y-x)+
  Heaviside(y)*Heaviside(x))+(x-1)*(-Heaviside(y+x)*Heaviside(y)*
  Heaviside(x)+Heaviside(y+x)*Heaviside(x))), x = -1 .. 1, y = -1 .
  . 1));
```

$$\mathit{Int}\left(-\mathrm{Heaviside}(y)\,\mathrm{Heaviside}(x)\,(x-1)^2\,(\mathrm{Heaviside}(y-x)-1), x=-1..1, y=-1..1\right)$$

An improvement in combine regarding linear functions:

```
> combine(Sum(Sum(a[u]*a[v], u = 1 .. n), v = 1 .. n) + Sum(Sum(a
  [u]*b[v], u = 1 .. n), v = 1 .. n) + Sum(Sum(a[u]*b[v], v = 1 ..
  n), u = 1 .. n));
```

$$\sum_{v=1}^{n}\sum_{u=1}^{n}\left(a_u a_v + 2\,a_u b_v\right)$$

Simplification of intat and Intat has been improved:

```
> simplify(intat(f(x),x=y)+Intat(-f(x),x=y));
```

$$0$$

Repeated application of simplify with this input previously failed to converge. Now it does:

```
> simplify(Intat(exp(Intat(g(_a),_a=_b)), _b=t)+Intat(h(_a),_a=t));
```

$$\int^{t} \mathrm{e}^{\int^{\_b} g(\_a)\,\mathrm{d}\_a}\,\mathrm{d}\_b + \int^{t} h(\_a)\,\mathrm{d}\_a$$

Simplification of logarithms is now more careful not to expand unnecessarily. This led to the following new result:

```
> simplify(ln(csgn(I*(x^(3/2)+1)))) assuming x, real;
```

$$\ln\left(\mathrm{csgn}\left(\mathrm{I}+\mathrm{I}x^{3/2}\right)\right)$$

There is a new formula in use when combining logarithms:

$\ln(a) - \ln(b) \quad \ln\left(\frac{a}{b}\right)$ when $arg(a) - arg(b) \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right]$

```
> ans := combine(ln(1-exp(-t)) - ln(1-exp(t))) assuming t > 0;
```

$$ans := -\ln\left(\frac{-1 + e^{t}}{-1 + e^{-t}}\right)$$

That led to an improved result from simplify as well:

```
> simplify(ans) assuming t > 0;
```

$$-t - I\pi$$

Improvement in simplification of radicals led to this new result from solve:

```
> solve(1-1/(1+203808*exp(-342569/506*t)*(1/131537))^
  (131537/203808)=44983/56599, t);
```

$$-\frac{506 \ln\left(\frac{7444862663\; 56599^{\frac{72271}{131537}}\; 11616^{\frac{59266}{131537}}}{27500110184448} - \frac{131537}{203808}\right)}{342569}$$

A weakness in simplification of piecewise functions regarding constant functions has been eliminated:

```
> simplify(piecewise(Or(And(f(0)<0, f(0)<=0), f(0)>=0), 1, 0));
```

$$1$$

simplify is now more careful when simplifying indexed piecewise functions:

```
> p1 := piecewise[undefined](t >= 200 and t <= 300, 1, t > 300 and
  t <= 400, 2):
  p2 := simplify(p1, piecewise);
  eval([p1,p2], t=100);
```

$$p2 := \begin{cases} undefined & t < 200 \\ 1 & t \leq 300 \\ 2 & t \leq 400 \\ undefined & 400 < t \end{cases}$$

$$[undefined, undefined]$$

There has been an improvement in simplify when simplifying constants:

```
> simplify((9*(20/(36+18*2^(1/2))^2+13/(36+18*2^(1/2))^2*2^(1/2))^
  (1/2)*x*2^(1/2)+9*(20/(
  36+18*2^(1/2))^2+13/(36+18*2^(1/2))^2*2^(1/2))^(1/2)*x+9*(20/
  (36+18*2^(1/2))^2+
  13/(36+18*2^(1/2))^2*2^(1/2))^(1/2)*y-252*(20/(36+18*2^(1/2))
  ^2+13/(36+18*2^(1/
  2))^2*2^(1/2))^(1/2)*z/(20+13*2^(1/2))-162*(20/(36+18*2^(1/2))
  ^2+13/(36+18*2^(1
  /2))^2*2^(1/2))^(1/2)*z/(20+13*2^(1/2))*2^(1/2)-31/9*(12*2^(1/2)
  +17)/(20/(36+18
  *2^(1/2))^2+13/(36+18*2^(1/2))^2*2^(1/2))^(1/2)/(23*2^(1/2)+33)/
  (8+2^(1/2)))^2-
  3844*(12*2^(1/2)+17)^2*(2+2^(1/2))^2/(20+13*2^(1/2))/(23*2^(1/2)
  +33)^2/(8+2^(1/
  2))^2+(9*(8/(36+18*2^(1/2))^2+1/(36+18*2^(1/2))^2*2^(1/2))^(1/2)*
  x-9*(8/(36+18*
  2^(1/2))^2+1/(36+18*2^(1/2))^2*2^(1/2))^(1/2)*y*2^(1/2)-9*(8/
  (36+18*2^(1/2))^2+
  1/(36+18*2^(1/2))^2*2^(1/2))^(1/2)*y+144*(8/(36+18*2^(1/2))^2+1/
  (36+18*2^(1/2))
  ^2*2^(1/2))^(1/2)*z/(8+2^(1/2))+126*(8/(36+18*2^(1/2))^2+1/
  (36+18*2^(1/2))^2*2^
  (1/2))^(1/2)*z/(8+2^(1/2))*2^(1/2)+31/9*(1+2^(1/2))/(23*2^(1/2)
  +33)/(8/(36+18*2
  ^(1/2))^2+1/(36+18*2^(1/2))^2*2^(1/2))^(1/2)/(8+2^(1/2)))^2-3844*
  (1+2^(1/2))^2*
  (2+2^(1/2))^2/(23*2^(1/2)+33)^2/(8+2^(1/2))^3-590*(3+2*2^(1/2))/
  (20+13*2^(1/2))
  /(8+2^(1/2))*(z-41/1180*(20+13*2^(1/2))*(10+7*2^(1/2))/(3+2*2^
  (1/2))/(23*2^(1/2
  )+33))^2+1681/2360*(20+13*2^(1/2))*(10+7*2^(1/2))^2/(3+2*2^(1/2))
  /(8+2^(1/2))/(
  23*2^(1/2)+33)^2-2);
```

$$5x^2 + (3y - 4z - 8)x + 2y^2 + (-6z - 4)y - 5z^2 + 7z - 2$$

## Mathematical Logic

Maple 2022 includes improvements to the the use of assume and is:

These all returned FAIL previously:

```
> is(x^2+x+1, odd) assuming x::integer;
```

$$\mathit{true}$$

```
> is(sec(x),real) assuming x::real, (x/Pi-1/2)::Non(integer);
```

$$\mathit{true}$$

```
> is(csc(x),real) assuming x::real, (x/Pi)::Non(integer);
```

$$\mathit{true}$$

Smarter handling of inequalities led to this new result from **couldItbe**:

```
> coulditbe(And(r^2=0,r*s<>0));
```

$$\mathit{false}$$

Some improvements in **is** relating to indexed variables resulted in this new result from **signum**:

```
> signum(r) assuming r>low[r],low[r]>0;
```

$$1$$

**is** and **coulditbe** can now handle more complicated RootOf functions:

```
> map(is, [`=`, `<`, `>`](RootOf (exp(_Z) - exp (1/5), 0.21),
  100000));
```

$$[\mathit{false}, \mathit{true}, \mathit{false}]$$

This led to the following new result from solve:

```
> solve({(sqrt(1+1/x)-ln(sqrt(1+x)+sqrt(x)))/sqrt(x)>1,x>0},{x});
```

$$\left\{0 < x, x < \frac{\left(\mathrm{e}^{2\,\mathit{RootOf}\left(\left(\mathrm{e}^{\_Z}\right)^{4}+2\left(\mathrm{e}^{\_Z}\right)^{3}\_Z-2\left(\mathrm{e}^{\_Z}\right)^{3}-2\left(\mathrm{e}^{\_Z}\right)^{2}-2\,\_Z\,\mathrm{e}^{\_Z}-2\,\mathrm{e}^{\_Z}+1,\,0.7505612007\right)}-1\right)^{2}}{4\left(\mathrm{e}^{\mathit{RootOf}\left(\left(\mathrm{e}^{\_Z}\right)^{4}+2\left(\mathrm{e}^{\_Z}\right)^{3}\_Z-2\left(\mathrm{e}^{\_Z}\right)^{3}-2\left(\mathrm{e}^{\_Z}\right)^{2}-2\,\_Z\,\mathrm{e}^{\_Z}-2\,\mathrm{e}^{\_Z}+1,\,0.7505612007\right)}\right)^{2}}\right\}$$

And also an improved result from Student:-Calculus1:-InflectionPoints:

```
> Student:-Calculus1:-InflectionPoints((1+sin(x))^cos(x), x = 0 ..
  2*Pi);
```

```
Warning, some inflection points are returned as numeric 
approximations
```

$$\left[0, 1.374838238, 3.780861806, \frac{3\,\pi}{2}, 2\,\pi\right]$$

This new result from **expand** is due to an improvement in **is**:

```
> expand(exp((sin(x + arctan(x*sqrt(2)/sqrt(-2*x^2 + 1)))^2)/4))
  assuming x > 0;
```

$$\left(\mathrm{e}^{\frac{\sin(x)^2}{1+\frac{2\,x^2}{-2\,x^2+1}}}\right)^{1/4} \mathrm{e}^{\frac{\sin(x)\cos(x)\,x\sqrt{2}}{2\left(1+\frac{2\,x^2}{-2\,x^2+1}\right)\sqrt{-2\,x^2+1}}} \sqrt{\mathrm{e}^{\frac{\cos(x)^2\,x^2}{\left(-2\,x^2+1\right)\left(1+\frac{2\,x^2}{-2\,x^2+1}\right)}}}$$

There is now more consistent error handling in assume:

```
> assume(Not(true), n=0);
  assume(false, n=0);
  assume(n=0, false);
```

```
Error, (in assume) contradictory assumptions
Error, (in assume) contradictory assumptions
Error, (in assume) contradictory assumptions
```

## Miscellaneous

This new result from minimize was due to improvement in **min** and **max**:

```
> alias(alpha=RootOf(2*_Z^2-2*RootOf(2*_Z^3-1,index = 1)-2*_Z+1,
  1.237360513)):
  minimize((x^2+y^2)^2+1/(x^2+y^2)+(x^2-2*x+1+y^2)^2+1/(x^2-2*x+1+
  y^2));
  alias(alpha=alpha):
```

$$\frac{8\,\alpha^6-24\,\alpha^5+36\,\alpha^4-32\,\alpha^3+18\,\alpha^2-6\,\alpha+9}{2\left(2\,\alpha^2-2\,\alpha+1\right)}$$

An improvement in allvalues led to a new result:

```
> {allvalues(RootOf(product(f(r, _Z), r=RootOf(_Z^2-1))))};
```

$$\{\mathit{RootOf}(f(-1, \_Z)), \mathit{RootOf}(f(1, \_Z))\}$$

Improved handling of dummy variables by depends:

```
> depends(sum(f[x], x=0..infinity), f[x]);
  depends(Product(f(x,a),x = 6 .. 5),a);
  depends(Product(f(x,a),x = 5 .. 5),a);
```

$$\mathit{false}$$
$$\mathit{false}$$
$$\mathit{true}$$

## Introduction of Ball Arithmetic

Maple 2022 includes new facilities for ball arithmetic in the form of RealBox and ComplexBox objects that provide an interface to the arblib package.

- The RealBox and ComplexBox objects represent boxes on the real line and in the complex plane, respectively.
- A RealBox object is defined by a center and radius. Thus, it represents a closed and bounded interval on the line, unless it is the entire real line.
- A ComplexBox object is defined as a pair of RealBox objects representing its real and imaginary parts. This represents either a closed and bounded box in the complex plane, the entire plane, or a doubly infinite strip in the direction of either the real or imaginary axis.
- Unlike the extant facilities in Maple for interval arithmetic, there is no need for you to use a special evaluator like evalr, since the arithmetic and function evaluations are built in to the real and complex box objects.
- The accuracy of box calculations is controlled by a **precision** option that is available for most operations, including the creation of real and complex boxes. This also affects the conversion of numeric quantities (centers and radii) in the construction of box objects.

```
> Radius( RealBox( 2/3 ) );
```

$$5.820766113350852282417152 \times 10^{-11}$$

Note that the radius for an inexactly representable quantity is smaller with greater precision, as directed by the **precision=80** option.

```
> Radius( RealBox( 2/3, 'precision' = 80 ) );
```

$$8.2718061697 \times 10^{-25}$$

```
> Radius( RealBox( 3/2 ) );
```

$$0.$$

Arithmetic operations on RealBox objects are built-in. You can operate with RealBox objects in many respects as though they were numeric quantities.

```
> p := randpoly( x );
```

$$p := -7\,x^5 + 22\,x^4 - 55\,x^3 - 94\,x^2 + 87\,x - 56$$

```
> eval( p, x = RealBox( Zeta( 3 ) ) );
                    ⟨RealBox: -154.41 ± 1.71822e-07⟩
```

```
> b := RealBox( 0.42, 1e-5 );
                    b := ⟨RealBox: 0.42 ± 1.00000e-05⟩
```

Many mathematical functions are defined for RealBox objects.

```
> sin( b );
                    ⟨RealBox: 0.40776 ± 9.13188e-06⟩
```

```
> Chi( b^2 );
                    ⟨RealBox: -1.15 ± 4.83663e-05⟩
```

For more information see, for example, RealBox/Circular, RealBox/Hyperbolic and RealBox/Special.

A ComplexBox object can be constructed from a complex numeric input.

```
> b := ComplexBox( 2.3 + 7.77*I, 'precision' = 50 );
          b := ⟨ComplexBox: [2.3 +/- 3.55271e-15] + [7.77 +/- 7.10543e-15]·I⟩
```

The result is an object with two RealBox components representing the real and imaginary parts.

```
> Re( b );
                    ⟨RealBox: 2.3 ± 3.55271e-15⟩
```

```
> Im( b );
                    ⟨RealBox: 7.77 ± 7.10543e-15⟩
```

As with RealBox objects, ComplexBox objects can be manipulated using ordinary arithmetic operations.

```
> eval( p, x = b );
          ⟨ComplexBox: [-176235 +/- 0.000143251] + [-102440 +/- 7.15837e-05]·I⟩
```

```
> 1/b^10 - I;
          ⟨ComplexBox: [7.90907e-10 +/- 1.20159e-19] + [-1 +/- 1.16415e-10]·I⟩
```

Apart from cases where the output of an operation is necessarily real, the result of a function applied to a ComplexBox is again a ComplexBox object.

```
> Li( b );
          ⟨ComplexBox: [3.76298 +/- 7.32189e-09] + [4.33322 +/- 6.45855e-09]·I⟩
```

```
> exp( b );
          ⟨ComplexBox: [0.836664 +/- 2.13816e-10] + [9.93903 +/- 2.44814e-09]·I⟩
```

Note that the magnitude of **b** is evaluated as a RealBox object.

```
> abs( b );
                    ⟨RealBox: 8.10326 ± 9.76226e-10⟩
```

For more information see, for example, ComplexBox/Circular, ComplexBox/Hyperbolic and ComplexBox/Elementary.

Certain specific constants, such as $\pi$, can be computed to high accuracy by using the **precision** option for the RealBox or ComplexBox constructor.

```
> RealBox( Pi );
                              ⟨RealBox: 3.14159 ± 1.16415e-10⟩
> RealBox( Pi, 'precision' = 100000 );
                              ⟨RealBox: 3.14159 ± 2.00495e-3010⟩
```

## Formal Power Series

The convert(...,FormalPowerSeries) command has been completely rewritten for Maple 2022. It is now able to find both closed form solutions and recurrence relations for many more examples than in previous versions.

## Graph Theory

Several new commands were added to the GraphTheory package.

- IsChordal tests if a graph is chordal.
- IsComparabilityGraph tests if a graph is a comparability graph.
- IsIntervalGraph tests if a graph is an interval graph.
- IsOriented tests if a directed graph is oriented, i.e. if it contents no pair of symmetric arcs.
- IsPerfectGraph tests if a graph is a perfect graph.
- IsSimplicial tests if a graph vertex is simplicial, i.e. if its neighborhood forms a clique.
- RichClubCoefficients computes the rich club coefficients for a graph. These are intended to measure the extent to which well-connected vertices also connect to each other.
- Traverse returns either a vertex list corresponding to a breadth-first or depth-first traversal of the graph, or an Iterator which can perform such a traversal interactively.

Additionally, within the SpecialGraphs subpackage, two related new commands were added.

- CompleteKaryArborescence computes the compute k-ary arborescence to a given depth. An arborescence is a directed analog of a tree with arcs leading away from the source.
- CompleteKaryAntiArborescence computes the compute k-ary anti-arborescence to a given depth. An anti-arborescence is a directed analog of a tree with arcs leading towards the sink.

## Linear Recurrence Relations

A new command, mhypergeomsols, was added to the **LREtools** package, which can find m-fold (also called m-interlacing) solutions of linear recurrence equations with polynomial coefficients for arbitrary positive integers $m$.

```
> with(LREtools);
```

$$\left[\mathit{AnalyticityConditions}, \mathit{GCRD}, \mathit{GeneralizedExponents}, \mathit{GuessRecurrence}, \mathit{HypergeometricTerm}, \mathit{IsDesingularizable}, \mathit{LCLM}, \mathit{MinimalRecurrence}, \mathit{MultiplyOperators}, \mathit{OperatorToRecurrence}, \mathit{REcontent}, \mathit{REcreate}, \mathit{REplot}, \mathit{REprimpart}, \mathit{REreduceorder}, \mathit{REtoDE}, \mathit{REtodelta}, \mathit{REtoproc}, \mathit{RecurrenceToOperator}, \mathit{RightDivision}, \mathit{RightFactors}, \mathit{SumDecompose}, \mathit{ValuesAtPoint}, \mathit{autodispersion}, \mathit{constcoeffsol}, \mathit{dAlembertiansols}, \delta, \mathit{dispersion}, \mathit{divconq}, \mathit{firstlin}, \mathit{hypergeomsols}, \mathit{mhypergeomsols}, \mathit{polysols}, \mathit{ratpolysols}, \mathit{riccati}, \mathit{shift}\right]$$

```
> RE := (n+1)*(n+2)*(n+3)*(n+4)*a(n+4) + 10*(n+1)*(n+2)*a(n+2) + 9*a
  (n) = 0;
```

$$\mathit{RE} := (n+1)\,(n+2)\,(n+3)\,(n+4)\,a(n+4) + 10\,(n+1)\,(n+2)\,a(n+2) + 9\,a(n) = 0$$

```
> hypergeomsols(RE, a(n), {}, output=basis);
```

$$\left[\frac{\mathrm{I}^n}{\Gamma(n+1)}, \frac{(-\mathrm{I})^n}{\Gamma(n+1)}, \frac{(-3\,\mathrm{I})^n}{\Gamma(n+1)}, \frac{(3\,\mathrm{I})^n}{\Gamma(n+1)}\right]$$

```
> msols := mhypergeomsols(RE, a(n), complex);
```

$$\mathit{msols} := \left[\left[1, \left\{\frac{\mathit{RootOf}\left(\_Z^2+1\right)^n}{n!}, \frac{\mathit{RootOf}\left(\_Z^2+9\right)^n}{n!}\right\}\right], \left[2, \left\{\frac{(-9)^n}{(2\,n)!}, \frac{(-1)^n}{(2\,n)!}\right\}\right]\right]$$

$\mathit{msols}[1]$ is equivalent to the solutions returned by **hypergeomsols**, while $\mathit{msols}[2]$ gives all 2-fold hypergeometric solutions, which cannot be found by **hypergeomsols**.

```
> allvalues(msols[1,2,1]), allvalues(msols[1,2,2]);
```

$$\frac{\mathrm{I}^n}{n!}, \frac{(-\mathrm{I})^n}{n!}, \frac{(3\,\mathrm{I})^n}{n!}, \frac{(-3\,\mathrm{I})^n}{n!}$$

```
> sol1 := unapply(eval(op([2,2,1],msols),n=n/2),n);
```

$$\mathit{sol1} := n \quad \frac{(-9)^{\frac{n}{2}}}{n!}$$

```
> normal(expand(eval(RE,a=sol1)));
```

$$0 = 0$$

```
> sol2 := unapply(eval(op([2,2,2],msols),n=n/2),n);
```

$$\mathit{sol2} := n \quad \frac{(-1)^{\frac{n}{2}}}{n!}$$

```
> normal(expand(eval(RE,a=sol2)));
```

$$0 = 0$$

## Polyhedral Sets

A new command, IntegerHull, was added to the PolyhedralSets package. It computes the integer hulls of 2-D (bounded or unbound) and 3-D (bounded) polyhedral sets.

The integer hull of a convex polyhedral set is the smallest polyhedral set that contains all the integer points in the input set.

> $restart; with(PolyhedralSets):$

> $ineqs := [2x + 5y \le 64, 7x + 5y \ge 20, 3x - 6y \le -7]:$

> $poly2d := PolyhedralSet(ineqs, [x, y]);$

$$poly2d := \begin{cases} Coordinates & : \quad [x, y] \\ Relations & : \quad \left[-x - \frac{5y}{7} \le -\frac{20}{7}, x - 2y \le -\frac{7}{3}, x + \frac{5y}{2} \le 32\right] \end{cases}$$

The input to the command is a PolyhedralSet object. By default, the command returns the integer hull in the form [vertices, rays].

> $IntegerHull(poly2d);$

$$[[[12, 8], [-8, 16], [-7, 14], [-5, 11], [0, 4], [1, 3], [3, 3], [11, 7]], [\ ]]$$

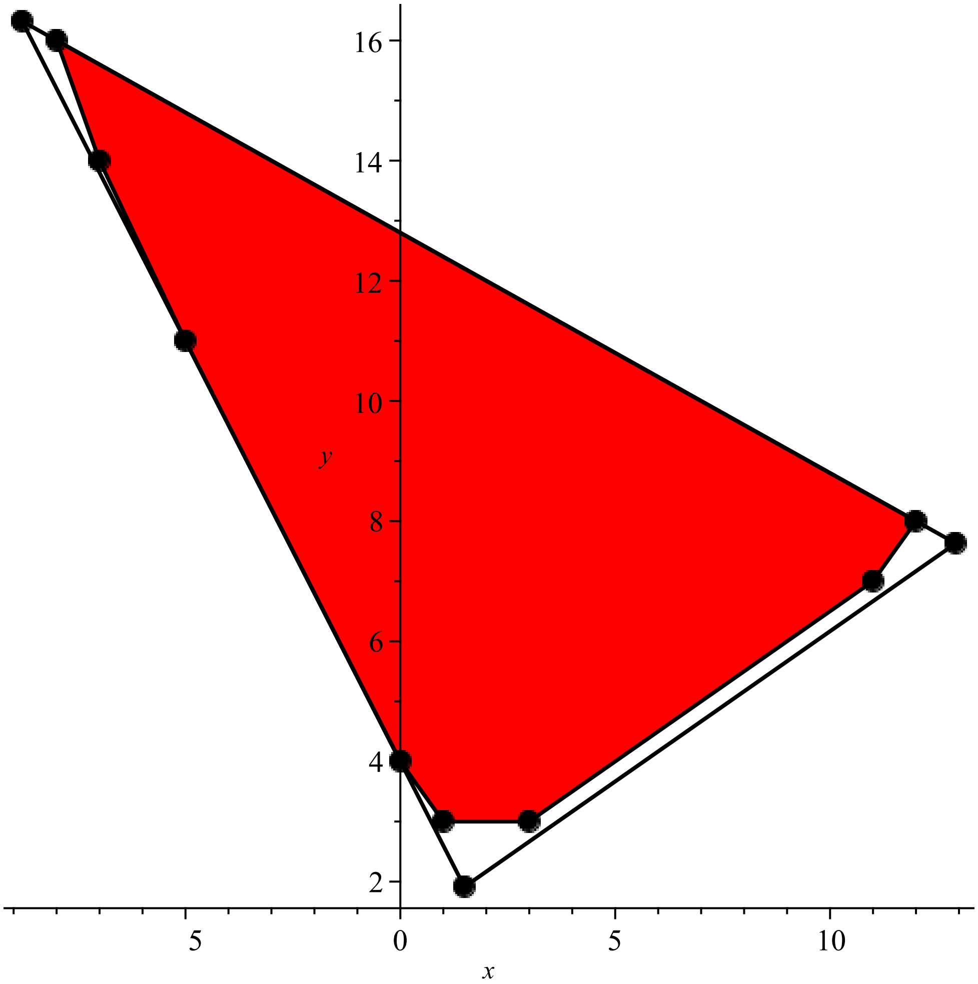
The command also has the option to return the integer hull in the PolyhedralSet object format.

> $intpoly2d := IntegerHull(poly2d, output = polyhedralset);$

$$intpoly2d := \begin{cases} Coordinates & : \quad [x, y] \\ Relations & : \quad \left[-y \le -3, -x - y \le -4, -x - \frac{5y}{7} \le -\frac{20}{7}, -x - \frac{2y}{3} \le -\frac{7}{3}, -x - \frac{y}{2} \le 0, x \right. \end{cases}$$

In this example, the white triangle is the input polyhedral set, and the red polygon is the output integer hull.

> $Plot([intpoly2d, poly2d], faceoptions = [transparency = [0, 0.13], color = [red, white]]);$



The input 2-D polyhedral set can be unbounded.

> $inset := PolyhedralSet([[\ \ 7/5, 0], [13/4, 0], [\ \ 5/3,\ \ 1], [5/2,\ \ 1]], [[0,\ \ 1]], [x, y]);$

$$inset := \begin{cases} Coordinates & : & [x, y] \\ Relations & : & \left[y \le 0,\quad x \le \frac{5}{3},\quad x + \frac{4\,y}{15} \le \frac{7}{5}, x \le \frac{13}{4}\right] \end{cases}$$

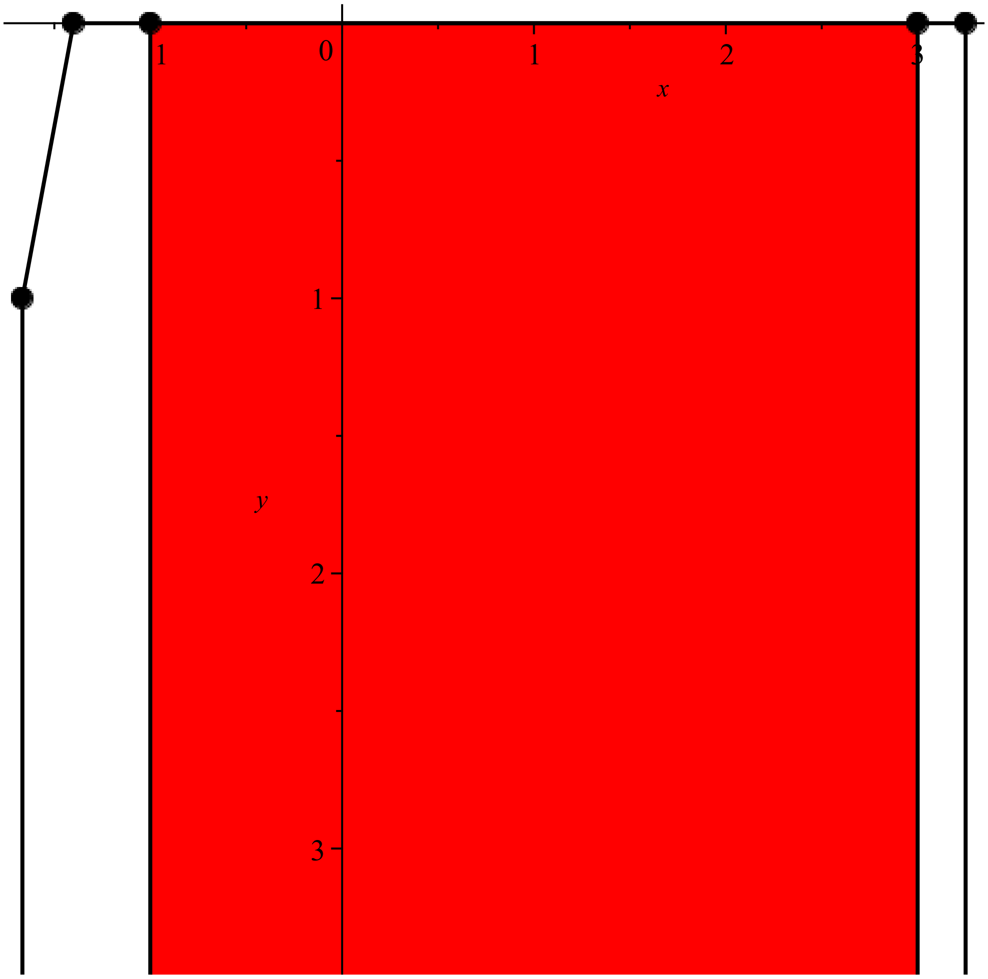
> $vert := IntegerHull(inset);$

$$vert := [[[-1, 0], [3, 0]], [[0, -1]]]$$

> $intpoly := PolyhedralSet(op(vert), [x, y]);$

$$intpoly := \begin{cases} Coordinates & : & [x, y] \\ Relations & : & [y \le 0,\quad x \le 1, x \le 3] \end{cases}$$

> $Plot([intpoly, inset], faceoptions = [transparency = [0, 0.13], color = [red, white]]);$



The following is a bounded 3-D example.

> $vars := [x1, x2, x3]:$

> $vertices := \left[\left[2, \frac{5}{2}, 3\right], [\quad 1, \quad 5/2, 3], \left[\quad \frac{3}{2}, 4, \quad 5/2\right], [5, \quad 5/2, \quad 4]\right]:$

> $poly := PolyhedralSet(vertices, [\ ], vars);$

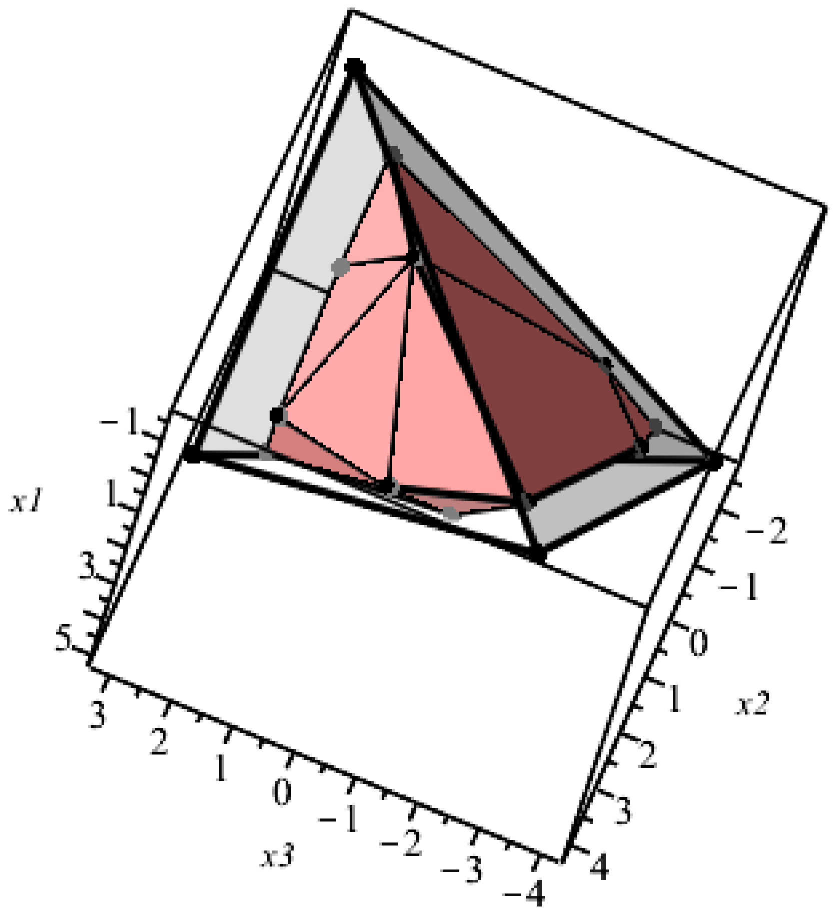
$$poly := \left\{\begin{array}{lll} Coordinates & : & [x1, x2, x3] \\ Relations & : & \left[\quad x1 \quad \frac{73\,x2}{91} \quad \frac{6\,x3}{7} \leq \frac{79}{182}, \quad x1 + \frac{3\,x2}{5} + \frac{4\,x3}{5} \leq \frac{19}{10}, x1 \quad \frac{3\,x2}{5} + \frac{6\,x3}{7} \leq \right. \end{array}\right.$$

> $intpoly := IntegerHull(poly, output = polyhedralset);$

$$intpoly := \left\{\begin{array}{lll} Coordinates & : & [x1, x2, x3] \\ Relations & : & \left[x3 \leq 2, \quad x2 \leq 2, x2 \leq 3, x2 + \frac{x3}{2} \leq 3, \quad x1 \quad x2 \quad x3 \leq 1, \quad x1 \quad \frac{7\,x2}{8} \quad \frac{3\,x3}{4} \leq \right. \end{array}\right.$$

In this example, the white transparent tetrahedron is the input polyhedral set, and the red polyhedron is the output integer hull.

> $Plot([intpoly, poly], faceoptions = [transparency = [0, 0.5], color = [red, white]]);$



The following shows a more complex example, with a higher number of vertices in the output.

> $vertices := \left[\left[\frac{15}{4}, -\frac{17}{2}, 5\right], [4, 8, 6], [-1, -10/3, 14], [-6/2, 8, -10/3], [14/2, -22/3, -4]\right]:$

> $poly := PolyhedralSet(vertices, [\ ], vars);$

$$poly := \begin{cases} Coordinates & : \quad [x1, x2, x3] \\ Relations & : \quad \left[ -x1 - \frac{1581\, x2}{2498} - \frac{1107\, x3}{2498} \leq -\frac{732}{1249}, -x1 - \frac{903\, x2}{1724} - \frac{783\, x3}{3448} \leq -\frac{747}{1724}, -x1 + \right. \end{cases}$$

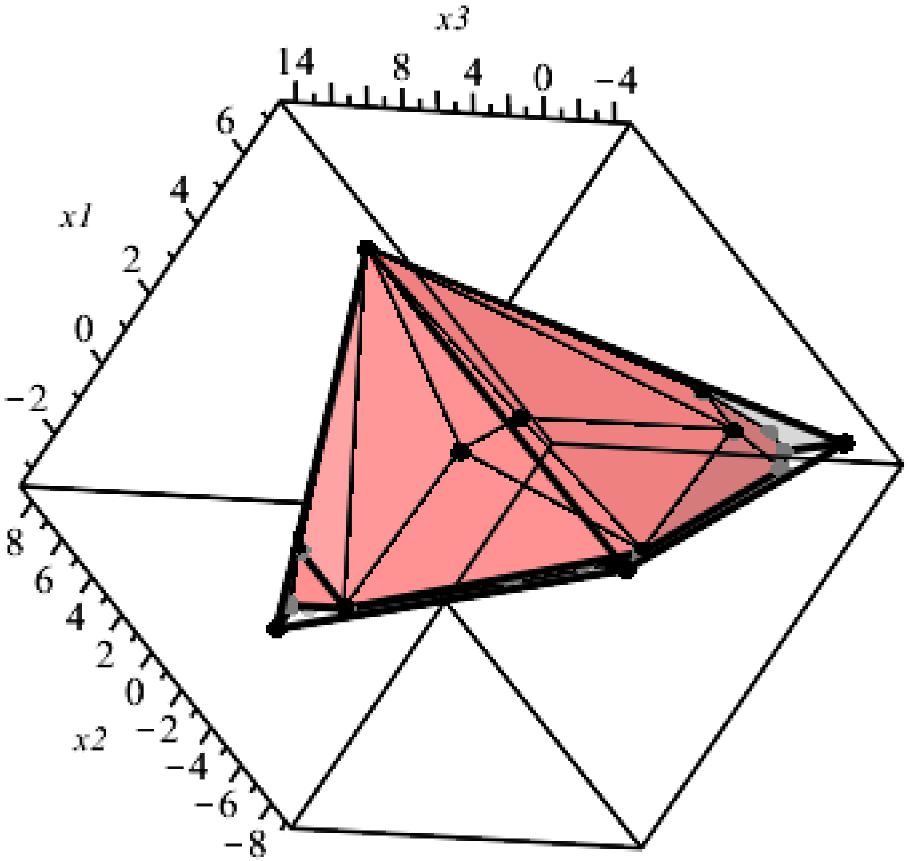
> $vert := IntegerHull(poly);$

$vert := [[[-2, 3, 4], [-2, 4, 3], [-2, 6, -2], [-2, 8, -2], [-1, -2, 11], [-1, -2, 12], [-1, 4, -2],$
$[-1, 5, -3], [0, -4, 12], [0, -1, 12], [0, 0, 2], [0, 0, 11], [0, 1, 0], [2, -6, 7], [2, 0, -3], [2, 1,$
$-3], [3, -7, 5], [3, -1, 7], [4, -8, 4], [4, -6, 1], [4, -1, 5], [4, 8, 6], [5, 1, 1], [6, -7, -2], [6,$
$-6, -3], [6, -5, -3], [6, -5, -1], [6, -3, -1]], [\ ]]$

> *intpoly* := *PolyhedralSet*(*op*(*vert*), *vars*);

$$intpoly := \begin{cases} Coordinates & : \quad [x1, x2, x3] \\ Relations & : \quad \left[-x3 \leq 3, x3 \leq 12, -x2 - x1 \leq 4, -x1 - \frac{2\,x2}{3} - \frac{2\,x3}{3} \leq 0, -x1 - \frac{2\,x2}{3} - \frac{4\,x3}{9} \leq \right. \end{cases}$$

> *Plot*([*intpoly*, *poly*], *faceoptions* = [*transparency* = [0, 0.5], *color* = [*red*, *white*]]);



The following tables are benchmarks of the new algorithm.

**Integer hulls of triangles (2-D)**

| | | | |
|---|---:|---:|---:|
| Volume | 27.95 | 111.79 | 11179.32 |
| Time(s) | 0.172 | 0.244 | 0.159 |
| Number of vertices in the output | 4 | 8 | 10 |

**Integer hulls of hexagons (2-D)**

| | | | |
|---|---:|---:|---:|
| Volume | 58.21 | 5820.95 | 23283.82 |
| Time(s) | 0.303 | 0.275 | 0.304 |
| Number of vertices in the output | 8 | 20 | 25 |

**Integer hulls of tetrahedrons (3-D)**

| | | | |
|---|---:|---:|---:|
| Volume | 447.48 | 6991.89 | 55935.2 |
| Time(s) | 0.977 | 1.223 | 1.378 |
| Number of vertices in the output | 27 | 33 | 39 |

# Real Root Finding

Two new commands were added to the RootFinding package.

- RefineRoot refines an isolating box for a real root of a polynomial system to arbitrary precision.
- EvaluateAtRoot evaluates a number of polynomials or constraints at such an isolating box, to arbitrary precision, or to ensure signs or truth values, respectively.

These two commands offer new ways of working with real roots for real polynomials at specified precisions.

> $with(RootFinding):$

RefineRoot allows for refinement of a real root:

> $p := x^3 - x - 2:$

> $rt := op([1, 2], Isolate(p, 'digits' = 15, 'output = interval'))$

$$rt := \left[\frac{449032033445746112207}{295147905179352825856}, \frac{224516016722873056117}{147573952589676412928}\right]$$

> $evalf(rt, 20)$

$$[1.5213797068045675696, 1.5213797068045675697]$$

> $refined := RefineRoot(rt, p, x, \text{'digits'} = 40)$

$$refined := \left[\frac{2120493824478992964488004125412730020651311}{1393796574908163946345982392040522594123776}, \frac{1060246912239496482244002062706365010325669}{696898287454081973172991196020261297061888}\right]$$

> $evalf(refined, 45)$

$$[1.52137970680456756960408083225443851442837514,\ 1.52137970680456756960408083225443851442839451]$$

RefineRoot is compatible with syntaxes previously and widely used within RootFinding, as well as real algebraic numbers represented by RootOfs:

> $RefineRoot\left(RootOf\left(\_Z^2 - 2\,RootOf\left(\_Z^3 - 9, 2\,..3\right), 1\,..2\right)\right)$

$$RootOf\left(\_Z^2 - 2\,RootOf\left(\_Z^3 - 9, \frac{4911459063769832938769}{2361183241434822606848} \,..\, \frac{19645836255079331755077}{9444732965739290427392}\right), \frac{9631969614722211425099}{4722366482869645213696} \,..\, \frac{19263939229444422850199}{9444732965739290427392}\right)$$

EvaluateAtRoot allows for evaluation of polynomials or relations on such at a real root. In particular, in taking in specific information about the real root, the routine provides robust evaluation of polynomial expressions in lieu of other evaluation methods involving *evalf* or similar.

EvaluateAtRoot offers a wide range of overloads to support various use cases amongst evaluation of constraints or polynomials.

In the case of relations, the main available output is a truth value:

> $constraints := \left[x^2 - 3 < 0, x^2 - 2 > 0, x + 4 > 0\right]:$

> $sys := \left[x^2 - 2\right]:$
$vars := [x]:$
$box := [1\,..2]:$

> $EvaluateAtRoot(constraints, box, sys, vars, \text{'digits'}= 30)$

$$[true, false, FAIL]$$

If the constraints are given as polynomials, the main available output is a list of signs:

> $polys := map(lhs - rhs, constraints)$

$$polys := \left[x^2 - 3, -x^2 + 2, -4 - x\right]$$

> $RootFinding\text{:-}EvaluateAtRoot(polys, box, sys, vars)$

$$[-1, 0, -1]$$

A multivariate example:

> $sys := [x^2+y+z-1, y^2+x+z-1, z^2+x+y-1]:$

> $vars := [x, y, z]:$

> $box := Isolate(sys, vars, 'output = interval', 'digits' = 20)$

$$box := \left[\left[x = \left[-\frac{7296512774041619012108 17}{30223145490365729367654 4}, -\frac{2918605109616647604843253}{1208925819614629174706176}\right], y = \left[-\frac{2918605109616647604843263}{1208925819614629174706176}, -\frac{2918605109616647604843255}{1208925819614629174706176}\right], z = \left[-\frac{1459302554808323802421631}{604462909807314587353088}, -\frac{2918605109616647604843257}{1208925819614629174706176}\right]\right], [x = [1, 1], y = [0, 0], z = [0, 0]], [x = [0, 0], y = [1, 1], z = [0, 0]], \left[x = \left[\frac{4006027763099114043447247}{967140655691703339764940 8}, \frac{2003013881549557021723633}{4835703278458516698824704}\right], y = \left[\frac{1001506940774778510861813}{2417851639229258349412352}, \frac{2003013881549557021723633}{4835703278458516698824704}\right], z = \left[\frac{1001506940774778510861813}{2417851639229258349412352}, \frac{2003013881549557021723629}{4835703278458516698824704}\right]\right], [x = [0, 0], y = [0, 0], z = [1, 1]]\right]$$

> $evalf(box[1], 25)$

$$[x = [-2.414213562373095048801696, -2.414213562373095048801684], y = [-2.414213562373095048801692, -2.414213562373095048801685], z = [-2.414213562373095048801691, -2.414213562373095048801687]]$$

> $constraints := [-17x^4y - 75x^3yz + 80x^2z^2 - 44y^3z + 71yz^3 - 82y^3 \geq 0, -62x^2z^3 + 97xy^3z - 73yz^4 - 56xyz^2 + 87xy < 0, -23xy^3z + 87xz^4 + 72x^2z^2 + 37xyz^2 + 74xy^2 + 6y^2 \neq 0]:$

> $EvaluateAtRoot(map(lhs - rhs, constraints), box[1], sys)$

$$[-1, 1, -1]$$

> $EvaluateAtRoot(constraints, box[1], sys)$

$$[true, false, FAIL]$$

> $values, refinedbox := EvaluateAtRoot(map(lhs - rhs, constraints), box[1], sys, 'accuracy' = 40, 'output = [intervals, refinement]'):$

```
> values
```

$$\left[\left[-\frac{4506467062523519726920791998043029943085529802413393}{3653754093327257295509212081790707549139831357 44},\right.\right.$$

$$\left.-\frac{4506467062523519726920791998043029943085529802413333}{365375409332725729550921208179070754913983135744}\right],$$

$$\left[\frac{5031029366959953335471150253290716055268453618081681}{292300327466180583640736966543256603931186508595 2},\right.$$

$$\left.\frac{5031029366959953335471150253290716055268453618082149}{2923003274661805836407369665432566039311865085952}\right],\left[\right.$$

$$-\frac{18650788634593516827581769937014533461857451017718 59}{730750818665451459101842416358141509827966271488},$$

$$\left.\left.-\frac{3730157726918703365516353987402906692371490203543559}{1461501637330902918203684832716283019655932542976}\right]\right]$$

```
> evalf(values, 45)
```

$$[[-12333.7995590714406028729662150027030187209087,$$

$$-12333.7995590714406028729662150027030187209087],$$

$$[1721.18499167334940947621998942265763945608181,$$

$$1721.18499167334940947621998942265763945608181],$$

$$[-2552.27748751002411421432998413398645918412272,$$

$$-2552.27748751002411421432998413398645918412272]]$$

```
> refinedbox
```

$$\left[x=\left[-\frac{14452232496229377357973232847790457004687879122145601}{5986310706507378352962293074805895248510699696029696},\right.\right.$$

$$\left.-\frac{14452232496229377357973232847790457004687879122145551}{5986310706507378352962293074805895248510699696029696}\right],y=\left[\right.$$

$$-\frac{3613058124057344339493308211947614251171969780536395}{1496577676626844588240573268701473812127674924007424},$$

$$\left.-\frac{14452232496229377357973232847790457004687879122145571}{5986310706507378352962293074805895248510699696029696}\right],z=\left[\right.$$

$$-\frac{14452232496229377357973232847790457004687879122145579}{5986310706507378352962293074805895248510699696029696},$$

$$\left.\left.-\frac{14452232496229377357973232847790457004687879122145573}{5986310706507378352962293074805895248510699696029696}\right]\right]$$

```
> evalf(refinedbox, 45)
[x = [-2.41421356237309504880168872420969807856967188,
    -2.41421356237309504880168872420969807856967188], y
    = [-2.41421356237309504880168872420969807856967188,
    -2.41421356237309504880168872420969807856967188], z
    = [-2.41421356237309504880168872420969807856967188,
    -2.41421356237309504880168872420969807856967188]]
```

# Intersection Multiplicities

A new algorithm has been implemented in the IntersectionMultiplicity command in the RegularChains package. It can handle cases that the previous version was unable to solve, and it is much faster in many cases.

## Summary

- With Maple 2022 comes a redesigned, more powerful version of the IntersectionMultiplicity command in the **AlgebraicGeometryTools** subpackage of the **RegularChains** package.
- The new command exceeds the previous version in both computational power and sheer performance. Using a newly designed algorithm, the new version of the **IntersectionMultiplicity** command can compute far more examples than previously possible. Moreover, the new version can compute these examples in a matter of seconds, far faster than the previous counterpart.
- The new algorithm incorporated into the **IntersectionMultiplicity** command no longer requires at least one polynomial to be non-singular at the point of interest. Hence, the intersection multiplicity command can now handle cases where all inputs are singular at the point of study.
- If the intersection multiplicity is required only at a single point, this can now be done directly, without having to encode the point as a regular chain.

## Examples

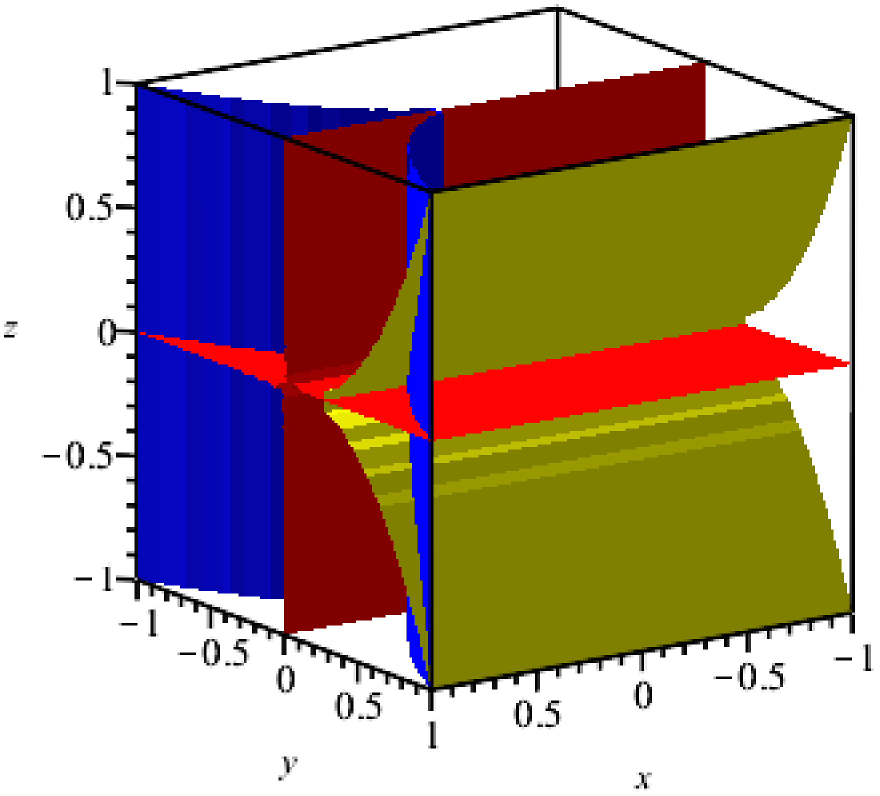
Here we see the polynomials in $F$ are all singular at the origin.

```
> with(RegularChains): with(ChainTools): with
  (AlgebraicGeometryTools);
[Cylindrify, IntersectionMultiplicity, IsTransverse, LimitPoints, RationalFunctionLimit,
    RegularChainBranches, TangentCone, TangentPlane, TriangularizeWithMultiplicity,
    consolidateTangentCones, partialTangentCone]
```

```
> F := [z*y, y^5-z^2, x^5-y^2];
```

$$F := \left[zy, y^5 - z^2, x^5 - y^2\right]$$

```
> plots:-implicitplot3d(F, x = -1 .. 1, y = -1 .. 1, z = -1 .. 1,
  numpoints = 10000, style = surface, color = ["red", "yellow",
  "blue"]);
```



Using the new version of the **IntersectionMultiplicity** command, we can compute the intersection multiplicity of this previously inaccessible example.

```
> IntersectionMultiplicity([0,0,0], F, [x,y,z]);
```

The previous version of the **IntersectionMultiplicity** command returns an error on this example. That version can still be accessed by passing **method=tangentcone**.

```
> R := PolynomialRing([x,y,z]):
```

```
> IntersectionMultiplicity(Chain([z,y,x],R), F, R, 'method=
  tangentcone');
```

```
Error, (in RegularChains:-AlgebraicGeometryTools:-
IntersectionMultiplicity) h top is singular
```

The new command is also easier to call when the input is a regular chain. It is no longer necessary to call Triangularize first and then map the **IntersectionMultiplicity** command to the output; the **IntersectionMultiplicity** command can now be called directly on regular chains encoding multiple points.

```
> F := [z, y, (x*(x+1))^2];
```

$$F := \left[z, y, x^2\,(x+1)^2\right]$$

```
> rc := Chain(F, R):
```

```
> res := IntersectionMultiplicity(rc, F, R);
```

$$res := [[2, regular\_chain], [2, regular\_chain]]$$

```
> Display(res, R);
```

$$\left[\left[2, \begin{cases} x = 0 \\ y = 0 \\ z = 0 \end{cases}\right], \left[2, \begin{cases} x+1 = 0 \\ y = 0 \\ z = 0 \end{cases}\right]\right]$$

The previous version of the **IntersectionMultiplicity** command returns a different error on this example.

```
> IntersectionMultiplicity(rc, F, R, 'method=tangentcone');
```

```
Error, (in RegularChains:-AlgebraicGeometryTools:-
IntersectionMultiplicity) this splits
```

This error can be avoided by first calling Triangularize on $F$ and then mapping **IntersectionMultiplicity** to the output. This extra step is inconvenient, and hence, the need for it has been eliminated in the updated command.

```
> dec := Triangularize(F, R);
```

$$dec := [regular\_chain, regular\_chain]$$

```
> res := map(chain -> IntersectionMultiplicity(chain, F, R,
  'method=tangentcone')[], dec);
```

$$res := [[2, regular\_chain], [2, regular\_chain]]$$

```
> Display(res, R);
```

$$\left[\left[2, \begin{cases} x+1=0 \\ y=0 \\ z=0 \end{cases}\right], \left[2, \begin{cases} x=0 \\ y=0 \\ z=0 \end{cases}\right]\right]$$

The new algorithm also offers significant improvements in performance, most notably as the size of the input system increases.

Consider the system $x_1, x_2^2, x_3,.., x_n$ where $n \geq 3$. When $n=3$ or $n=4$, the performance of the previous algorithm is about the same. When $n=5$ or $n=6$, however, the difference in run times becomes more pronounced. For $n \geq 10$, the difference in performance is dramatic.

```
> X := [seq(x[i], i = 1 .. 3)]:
```

```
> R := PolynomialRing(X):
```

```
> rc := Chain(ListTools:-Reverse(X), R):
```

```
> CodeTools:-Usage(IntersectionMultiplicity(rc, [x[1], x[2]^2, op
  (3..-1,X)], R));
memory used=1.03MiB, alloc change=0 bytes, cpu time=22.00ms, real
time=23.00ms, gc time=0ns
```

$$[[2, \mathit{regular\_chain}]]$$

```
> CodeTools:-Usage(IntersectionMultiplicity(rc, [x[1], x[2]^2, op
  (3..-1,X)], R, 'method=tangentcone'));
memory used=24.73MiB, alloc change=0 bytes, cpu time=449.00ms, real
time=450.00ms, gc time=0ns
```

$$[[2, \mathit{regular\_chain}]]$$

```
> X := [seq(x[i], i = 1 .. 6)]:
```

```
> R := PolynomialRing(X):
```

```
> rc := Chain(ListTools:-Reverse(X), R):
```

```
> CodeTools:-Usage(IntersectionMultiplicity(rc, [x[1], x[2]^2, op
  (3..-1,X)], R));
memory used=4.36MiB, alloc change=0 bytes, cpu time=84.00ms, real
time=85.00ms, gc time=0ns
```

$$[[2, \mathit{regular\_chain}]]$$

```
> CodeTools:-Usage(IntersectionMultiplicity(rc, [x[1], x[2]^2, op
  (3..-1,X)], R, 'method=tangentcone'));
memory used=398.46MiB, alloc change=0 bytes, cpu time=8.32s, real
time=7.90s, gc time=886.38ms
```

$$[[2, \mathit{regular\_chain}]]$$

```
> X := [seq(x[i], i = 1 .. 11)]:
> R := PolynomialRing(X):
> rc := Chain(ListTools:-Reverse(X), R):
> CodeTools:-Usage(IntersectionMultiplicity(rc, [x[1], x[2]^2, op
  (3..-1,X)], R));
memory used=28.53MiB, alloc change=0 bytes, cpu time=764.00ms, real
time=679.00ms, gc time=176.99ms
```

$$[[2, \mathit{regular\_chain}]]$$

```
> CodeTools:-Usage(IntersectionMultiplicity(rc, [x[1], x[2]^2, op
  (3..-1,X)], R, 'method=tangentcone'));
memory used=2.96GiB, alloc change=32.00MiB, cpu time=46.16s, real
time=43.68s, gc time=4.20s
```

$$[[2, \mathit{regular\_chain}]]$$

Of course, the system in this example is quite simple. For more complicated systems, the difference in performance can be seen even sooner.

Here, slightly modifying the first 3 polynomials has a large impact on the previous version of the intersection multiplicity command, while the performance of the new command remains relatively unchanged.

```
> X := [seq(x[i], i = 1 .. 6)]:
> R := PolynomialRing(X):
> rc := Chain(ListTools:-Reverse(X), R):
> CodeTools:-Usage(IntersectionMultiplicity(rc, [x[1]^2+x[2], x[2]
  ^2+x[3], x[3]^2+x[1]^2, op(4..-1,X)], R));
memory used=6.06MiB, alloc change=0 bytes, cpu time=96.00ms, real
time=95.00ms, gc time=0ns
```

$$[[2, \mathit{regular\_chain}]]$$

```
> CodeTools:-Usage(IntersectionMultiplicity(rc, [x[1]^2+x[2], x[2]
  ^2+x[3], x[3]^2+x[1]^2, op(4..-1,X)], R, 'method=tangentcone'));
```

```
memory used=2.40GiB, alloc change=0 bytes, cpu time=33.04s, real
time=31.00s, gc time=3.23s
```

$$[[2, \mathit{regular\_chain}]]$$