

Maple 16 の新規/拡張パッケージ

Maple 16 で新しく追加されたパッケージおよび拡張されたパッケージは次のとおりです。

[VectorCalculus](#) パッケージ

[RegularChains](#) パッケージ

[SolveTools\[SemiAlgebraic\]](#) コマンド

[Groebner\[IsBasis\]](#) コマンド

[Statistics](#) パッケージ

[ListTools](#) パッケージ

VectorCalculus

- [VectorCalculus\[Hessian\]](#) コマンドに、`determinant` オプションを使用できるようになりました。このオプションは、ヘシアン行列そのもののほか、入力関数の ヘシアン 行列の行列式を返すように要求します。

```
> with(VectorCalculus):
```

```
> H, det := Hessian( cos(x*y), [x,y], determinant);
```

$$H, det := \begin{bmatrix} -\cos(xy) y^2 & -\cos(xy) xy - \sin(xy) \\ -\cos(xy) xy - \sin(xy) & -\cos(xy) x^2 \\ -\sin(xy)^2 & \end{bmatrix}, -2 \cos(xy) xy \sin(xy) \quad (1)$$

RegularChains

- 新しい [RegularChains](#) パッケージは、効率アップや新しいアルゴリズムを導入するだけでなく、実数や準代数系の計算を行う新規および既存のコマンドも含まれます。
- 多項式系の可変次数は、三角形分解法の性能に大きな影響を及ぼします。新しいコマンド [SuggestVariableOrder](#) を使用すると、[RegularChains](#) ライブラリのソルバーで使用できる最適な次数を設定できます。
- 新しい [SquarefreeFactorization](#) コマンドを使用すると、レギュラーチェーンを法として多項式のべき乗の平方因子を含まない因数分解を計算できます。

```
> with(RegularChains):
```

```
> with(ChainTools):
```

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

```
> p := (x+y)*(x+2*y)^2*(x+3*y)^3:
```

```
> rc := Chain([y^2+1], Empty(R), R):
```

```
> p := SparsePseudoRemainder(p, rc, R):
```

```
> SquarefreeFactorization(p, x, rc, R);
```

```
[[[x + y, 1], [x + 2 y, 2], [x + 3 y, 3]], regular_chain]] \quad (2)
```

- [RealTriangularize](#) コマンドに 2 つの新しいオプション `method` および `relaxation` が追加

されました。 `method` オプションは、使用する分解アルゴリズムを指定します。 `relaxation` オプションは、新しい緩和手法を使用するかどうかを指定します。 `relaxation=explicit` に設定された場合、通常、出力コンポーネント数が削減されます。これは、厳密でない制約は不等式で使用できないためです。

```
> R := PolynomialRing([x, a, b]):
> sys := [x^5*a+x*b-a = 0, x-b > 0]:
> numelems(RealTriangularize(sys, R, output = list));
```

13 (3)

```
> numelems(RealTriangularize(sys, R, output = list, relaxation =
explicit));
```

7 (4)

```
> numelems(RealTriangularize(sys, R, output = list, method =
incremental));
```

13 (5)

- 新しい [RealComprehensiveTriangularize](#) コマンドは、パラメトリック準代数系の実数で三角分解を計算します。
- 新しい [LinearSolve](#) コマンドは、フーリエモツキンの非推奨アルゴリズムの改良型を実装するための線形準代数システムソルバーです。

```
> with(SemiAlgebraicSetTools):
> R := PolynomialRing([z, y, x]):
> F := [z+y+x = 0, 0 <= x+y-z, x-y-z < 0]:
> T := LinearSolve(F, R);
```

$T := [x < 0, -x \leq y, z = -x - y]$ (6)

- [CylindricalAlgebraicDecompose](#) コマンドで、入力として通常の準代数系のリストを使用できるようになりました。さらに、`output` オプションは出力フォーマットをコントロールし、`cadcell` と `rootof` を受け取ります。 [Display](#) コマンドは更新され、これらの新規出力フォーマットを扱います。
- [ConstructibleSetTools](#) のつぎのコマンドは、準代数系を入力として扱います。 [Complement](#)、 [Difference](#)、 [Intersection](#)、 [IsContained](#)、 [IsEmpty](#)、 [Projection](#) および [RemoveRedundantComponents](#)。

▼ SolveTools[SemiAlgebraic]

- [SolveTools\[SemiAlgebraic\]](#) コマンドは、方程式と不等式の組み合わせで構成される多項式の解を計算します。

```
> with(SolveTools):
> SemiAlgebraic({x^2+x-1=0, x+y^2<0});
```

$\left[\left[x = -\frac{1}{2} - \frac{1}{2} \sqrt{5}, -\sqrt{-x} < y, y < \sqrt{-x} \right] \right]$ (7)

▼ Groebner[IsBasis]

- [Groebner\[IsBasis\]](#) コマンドは、単項式順序について基底がグレブナ基底であるかどうかを判断するための便利な方法です。

```
> with(Groebner):
```

```
> F:=[d^5-e^5, c^5-d^5, b^5-c^5, a^4*b+b^4*c+c^4*d+d^4*e+a*e^4,a^5-b^5]:
```

```
> G:=Basis(F, tdeg(a,b,c,d,e)):
```

```
> IsBasis(G, tdeg(a,b,c,d,e));
```

true

(8)

```
> IsBasis(G, plex(a,b,c,d,e));
```

false

(9)

Statistics

- [Statistics](#) パッケージは、非整数値の離散型分布機能、離散型分布のカスタマイズ、効率的な最大尤度パラメータ推定、行列データセットの処理の向上、可変幅ヒストグラムなどの新機能、および lowess スムージング用の新規コマンド [ScatterPlot3D](#) が含まれています。これらの変更点の詳細については、[What's New, Statistics](#) を参照してください。

Student

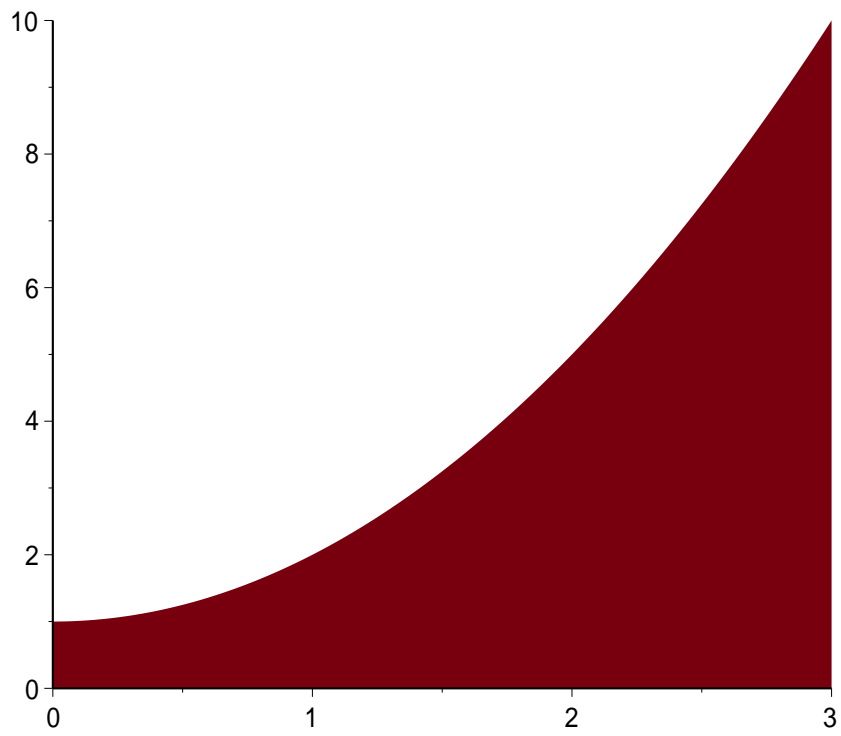
- [Student](#) サブパッケージの多くのコマンドは、[コマンドの補完機能](#)によりアクセス可能です。

- [Student\[Calculus1\]\[VolumeOfRevolution\]](#) コマンドは、新規の `output` オプション `region` を持ちます。新規の `output` オプションは、回転してソリッドを作成するスライスの可視化に使用できます。さらに、新規の `showregion` オプションは、ソリッド内にスライスを組み込むことができます。

```
> with(Student):
```

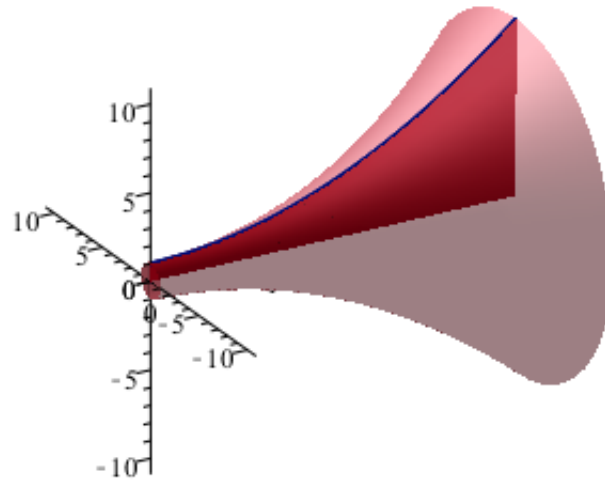
```
> with(Calculus1):
```

```
> VolumeOfRevolution(x^2+1, x=0..3, output=region);
```



The region used to create a solid of revolution on $0 \leq x \leq 3$ of
 $f(x) = x^2 + 1$ about the axis $y=0$.

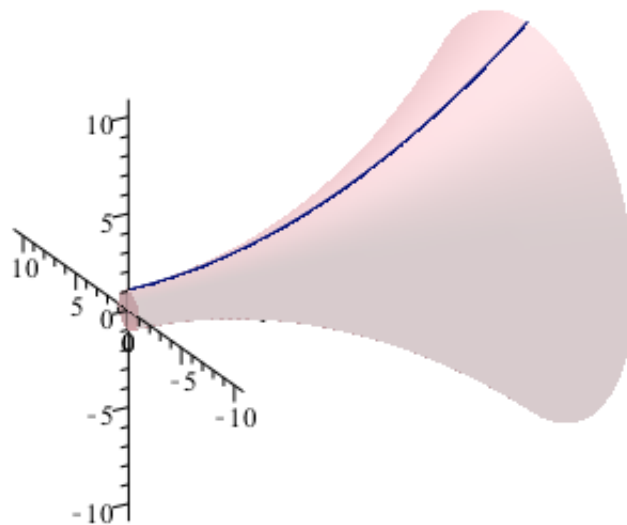
```
> VolumeOfRevolution(x^2+1, x=0..3, output=plot, showregion=true);
```



The solid of revolution created on $0 \leq x \leq 3$ by rotation of $f(x) = x^2 + 1$ about the axis $y=0$. The slice that is rotated is shaded in burgundy.

- 可視化ルーチンによって生成されるプロットのデフォルトのキャプションは、`caption` オプションを空文字列にすることにより無効にすることができます。

```
> VolumeOfRevolution(x^2+1, x=0..3, output=plot, caption="");
```



SCSCP

- [SCSCP](#) は新規パッケージで、[SCIENCE project](#) について Symbolic Computation Software Composability Protocol (SCSCP) を実装しています。[SCSCP\[Client\]](#) サブパッケージはリモートサーバーに計算の実行を要求し、数式をリモートで保存および取得し、サーバーのプロパティを問い合わせます。[SCSCP\[Server\]](#) サブパッケージは、計算の実行用にシンボリック計算ソフトウェアとして Maple を使用しているサーバーの設定に使用できます。
- サーバーの設定には MapleNET 15 およびそれ以降が必要です。

> with(SCSCP):

> with(Client):

> CallService("scscp.maplesoft.com", "scscp_transient_maple",
"UnivariateRootFinding", [(x^2-4)*(x^2-7)]);

$[x = -2.645751311, x = -2., x = 2., x = 2.645751311]$

(10)

Threads

- [Threads](#) パッケージの [Add](#)、[Mul](#)、[Seq](#) および [Map](#) ルーチンは更新され、特に入力が少ないときの並列処理の開発において強化されています。さらに、1回のタスクで実行される最大要素数を指定することができます。これにより、入力に見合うルーチンの実行を調整で

きます。たとえば、長時間実行される計算の回数が少ない場合は最大タスクサイズが 1 でも望ましい場合があります。しかし、非常に高速な計算が多くある場合は、タスクサイズが大きいことが望ましくなります。

▼ Magma

- [Magma](#) パッケージは、新規コマンド [IsCrossedSet](#)、[IsDiassociative](#)、[IsPowerAssociative](#)、[IsQuasitrivial](#)、[IsomorphicCopoly](#)、および [IsomorphismClassRepresentatives](#) を含んでいます。

```
> with( Magma );
```

```
> L := select(IsCrossedSet,Enumerate(3,quandle,output=list));
```

$$L := \left[\left[\begin{array}{ccc} 1 & 1 & 1 \\ 2 & 2 & 2 \\ 3 & 3 & 3 \end{array} \right], \left[\begin{array}{ccc} 1 & 3 & 2 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \end{array} \right] \right] \quad (11)$$

```
> select( IsSimple, L );
```

$$\left[\left[\begin{array}{ccc} 1 & 3 & 2 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \end{array} \right] \right] \quad (12)$$

▼ ListTools

- [ListTools](#) パッケージは新規コマンド [Classify](#) を含んでいます。これは、与えられたプロパティに従ってリスト要素をグループ化するための効率的な方法です。

```
> with(ListTools);
```

```
[BinaryPlace, BinarySearch, Categorize, Classify, DotProduct, Enumerate, FindMaximalElement, FindMinimalElement, FindRepetitions, Flatten, FlattenOnce, Group, Interleave, Join, JoinSequence, LengthSplit, MakeUnique, Occurrences, Pad, PartialSums, Reverse, Rotate, Search, SearchAll, SelectFirst, SelectLast, Sorted, Split, Transpose] \quad (13)
```

```
> types := Classify(whattype, ["Hello, 1, 2., Pi, x, x+y, exp, "world!"]):
```

```
Error, incorrect syntax in parse: missing operator or `:` (near 67th character of parsed string)
```

```
> eval(types);
```

types \quad (14)

▼ 参照

[Maple 16 新機能一覧](#)