

Differential geometry with SageMath

Éric Gourgoulhon

Laboratoire Univers et Théories (LUTH)
Observatoire de Paris, CNRS, Université PSL, Université Paris Cité
Meudon, France

<https://luth.obspm.fr/~luthier/gourgoulhon>

GDR Géométrie Différentielle et Mécanique

ENS Paris-Saclay
23-25 November 2022

- 1 SageMath and its differential geometry capabilities
- 2 SageMath implementation of tensor fields
- 3 Example: Schwarzschild spacetime
- 4 Other examples
- 5 Conclusions

Outline

- 1 SageMath and its differential geometry capabilities
- 2 SageMath implementation of tensor fields
- 3 Example: Schwarzschild spacetime
- 4 Other examples
- 5 Conclusions

SageMath in a few words

SageMath (*nickname: Sage*) is a **free open-source** computer algebra system started by **William Stein** in 2005

SageMath in a few words

SageMath (*nickname: Sage*) is a **free open-source** computer algebra system started by **William Stein** in 2005

SageMath is free (GPL v2)

Freedom means

- 1 everybody can use it, by download from <https://www.sagemath.org>
- 2 everybody can examine the source code and improve it

SageMath in a few words

SageMath (*nickname: Sage*) is a **free open-source** computer algebra system started by **William Stein** in 2005

SageMath is free (GPL v2)

Freedom means

- 1 everybody can use it, by download from <https://www.sagemath.org>
- 2 everybody can examine the source code and improve it

SageMath is based on Python

- no need to learn any specific syntax to use it
- Python is a powerful *object oriented language*, with a neat syntax
- SageMath benefits from the Python ecosystem (e.g. **Jupyter notebook**, **NumPy**, **Matplotlib**)

SageMath in a few words

SageMath (nickname: **Sage**) is a **free open-source** computer algebra system started by **William Stein** in 2005

SageMath is free (GPL v2)

Freedom means

- 1 everybody can use it, by download from <https://www.sagemath.org>
- 2 everybody can examine the source code and improve it

SageMath is based on Python

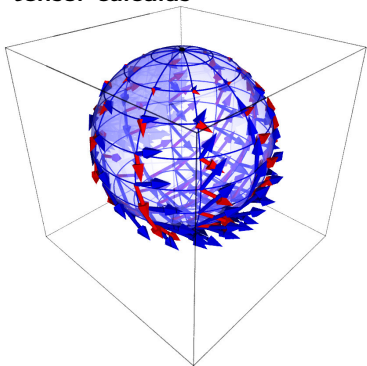
- no need to learn any specific syntax to use it
- Python is a powerful *object oriented language*, with a neat syntax
- SageMath benefits from the Python ecosystem (e.g. **Jupyter notebook**, **NumPy**, **Matplotlib**)

SageMath is developed by an enthusiastic community

- mostly composed of mathematicians
- welcoming newcomers

Differential geometry with SageMath

SageManifolds project: extends SageMath towards **differential geometry** and **tensor calculus**



Stereographic-coordinate frame on \mathbb{S}^2

- <https://sagemanifolds.obspm.fr>
- ~ 119,000 lines of Python code
- fully included in SageMath (after **review process**)
- ~ 30 contributors (developers and reviewers) cf. <https://sagemanifolds.obspm.fr/authors.html>
- dedicated **mailing list**
- help: <https://ask.sagemath.org>

Everybody is welcome to contribute

⇒ visit <https://sagemanifolds.obspm.fr/contrib.html>

Current status

Already present (SageMath 9.7):

- **differentiable manifolds**: tangent spaces, vector frames, tensor fields, curves, pullback and pushforward operators, submanifolds
- **vector bundles** (tangent bundle, tensor bundles)
- **standard tensor calculus** (tensor product, contraction, symmetrization, etc.), even on non-parallelizable manifolds, and with all **monoterm tensor symmetries** taken into account
- **Lie derivative** along a vector field
- **differential forms**: exterior and interior products, exterior derivative, Hodge duality
- **multivector fields**: exterior and interior products, Schouten-Nijenhuis bracket
- **affine connections** (curvature, torsion)
- **pseudo-Riemannian metrics**
- **computation of geodesics** (numerical integration)

Current status

Already present (*cont'd*):

- some **plotting capabilities** (charts, points, curves, vector fields)
- **parallelization** (on tensor components) of CPU demanding computations
- **extrinsic geometry** of pseudo-Riemannian submanifolds
- **series expansions** of tensor fields
- **symplectic manifolds**
- 2 symbolic backends: **Pynac/Maxima** (SageMath's default) and **SymPy**

Future prospects:

- more symbolic backends (Giac, FriCAS, ...)
- more graphical outputs
- spinors, integrals on submanifolds, variational calculus, etc.
- **connection with numerical relativity**: use SageMath to explore numerically-generated spacetimes

Outline

- 1 SageMath and its differential geometry capabilities
- 2 SageMath implementation of tensor fields
- 3 Example: Schwarzschild spacetime
- 4 Other examples
- 5 Conclusions

Vector fields on a smooth manifold

The set $\mathfrak{X}(M)$ of vector fields on a smooth manifold M over $\mathbb{K} = \mathbb{R}$ or $\mathbb{K} = \mathbb{C}$ is endowed with two algebraic structures:

- 1 $\mathfrak{X}(M)$ is an **infinite-dimensional vector space over \mathbb{K}** , the scalar multiplication $\mathbb{K} \times \mathfrak{X}(M) \rightarrow \mathfrak{X}(M)$, $(\lambda, \mathbf{v}) \mapsto \lambda \mathbf{v}$ being defined by

$$\forall p \in M, \quad (\lambda \mathbf{v})|_p = \lambda \mathbf{v}|_p,$$

- 2 $\mathfrak{X}(M)$ is a **module over the commutative algebra $C^\infty(M)$** , the scalar multiplication $C^\infty(M) \times \mathfrak{X}(M) \rightarrow \mathfrak{X}(M)$, $(f, \mathbf{v}) \mapsto f \mathbf{v}$ being defined by

$$\forall p \in M, \quad (f \mathbf{v})|_p = f(p) \mathbf{v}|_p,$$

the right-hand side involving the scalar multiplication by $f(p) \in \mathbb{K}$ in the vector space $T_p M$.

$\mathfrak{X}(M)$ as a $C^\infty(M)$ -module

$\mathfrak{X}(M)$ is a **free module** over $C^\infty(M) \iff \mathfrak{X}(M)$ admits a basis

If this occurs, then $\mathfrak{X}(M)$ is actually a **free module of finite rank** over $C^\infty(M)$ and $\text{rank } \mathfrak{X}(M) = \dim M = n$.

One says then that M is a **parallelizable** manifold.

A basis $(e_a)_{1 \leq a \leq n}$ of $\mathfrak{X}(M)$ is called a **vector frame**


Basis expansion¹:

$$\forall v \in \mathfrak{X}(M), \quad v = v^a e_a, \quad \text{with } v^a \in C^\infty(M) \quad (1)$$

At each point $p \in M$, (1) gives birth to an identity in the tangent space $T_p M$:

$$v|_p = v^a(p) e_a|_p, \quad \text{with } v^a(p) \in \mathbb{K},$$

which is nothing but the expansion of the tangent vector $v|_p$ on the basis $(e_a|_p)_{1 \leq a \leq n}$ of the vector space $T_p M$.

¹Einstein's convention for summation on repeated indices is assumed. 

Parallelizable manifolds

M is **parallelizable** \iff $\mathfrak{X}(M)$ is a free $C^\infty(M)$ -module of rank n
 \iff M admits a global vector frame
 \iff the tangent bundle is trivial: $TM \simeq M \times \mathbb{K}^n$

Parallelizable manifolds

M is **parallelizable** $\iff \mathfrak{X}(M)$ is a free $C^\infty(M)$ -module of rank n
 $\iff M$ admits a global vector frame
 \iff the tangent bundle is trivial: $TM \simeq M \times \mathbb{K}^n$

Examples of parallelizable manifolds

- \mathbb{R}^n (global coordinate chart \Rightarrow global vector frame)
- the circle \mathbb{S}^1 (rem: no global coordinate chart)
- the torus $\mathbb{T}^2 = \mathbb{S}^1 \times \mathbb{S}^1$
- the 3-sphere $\mathbb{S}^3 \simeq \text{SU}(2)$, as any Lie group
- the 7-sphere \mathbb{S}^7
- any orientable 3-manifold (Steenrod theorem)

Parallelizable manifolds

M is **parallelizable** $\iff \mathfrak{X}(M)$ is a free $C^\infty(M)$ -module of rank n
 $\iff M$ admits a global vector frame
 \iff the tangent bundle is trivial: $TM \simeq M \times \mathbb{K}^n$

Examples of parallelizable manifolds

- \mathbb{R}^n (global coordinate chart \Rightarrow global vector frame)
- the circle \mathbb{S}^1 (rem: no global coordinate chart)
- the torus $\mathbb{T}^2 = \mathbb{S}^1 \times \mathbb{S}^1$
- the 3-sphere $\mathbb{S}^3 \simeq \text{SU}(2)$, as any Lie group
- the 7-sphere \mathbb{S}^7
- any orientable 3-manifold (Steenrod theorem)

Examples of non-parallelizable manifolds

- the sphere \mathbb{S}^2 (hairy ball theorem!) and any n -sphere \mathbb{S}^n with $n \notin \{1, 3, 7\}$
- the real projective plane $\mathbb{R}\mathbb{P}^2$

SageMath implementation of vector fields

Choice of the $C^\infty(M)$ -module point of view for $\mathfrak{X}(M)$, instead of the infinite-dimensional \mathbb{K} -vector space one

⇒ implementation advantages:

- reduction to finite-dimensional structures: free $C^\infty(U)$ -modules of rank n on parallelizable open subsets $U \subset M$
- for tensor calculus on each parallelizable open set U , use of exactly the same `FiniteRankFreeModule` code as for the tangent spaces

SageMath implementation of vector fields

Choice of the $C^\infty(M)$ -module point of view for $\mathfrak{X}(M)$, instead of the infinite-dimensional \mathbb{K} -vector space one

⇒ implementation advantages:

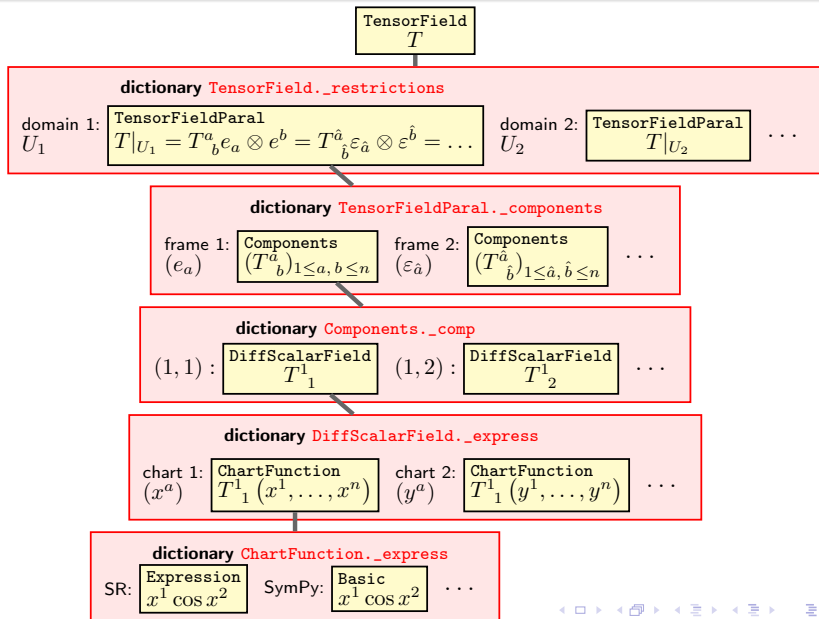
- reduction to finite-dimensional structures: free $C^\infty(U)$ -modules of rank n on parallelizable open subsets $U \subset M$
- for tensor calculus on each parallelizable open set U , use of exactly the same `FiniteRankFreeModule` code as for the tangent spaces

Decomposition of M into parallelizable parts

Assumption: the smooth manifold M can be covered by a finite number m of parallelizable open subsets U_i ($1 \leq i \leq m$)

Example: this holds if M is compact (finite atlas)

Tensor field storage



Outline

- 1 SageMath and its differential geometry capabilities
- 2 SageMath implementation of tensor fields
- 3 Example: Schwarzschild spacetime
- 4 Other examples
- 5 Conclusions

Pseudo-Riemannian manifolds in SageMath

The Schwarzschild spacetime example

The **Schwarzschild spacetime** is a 4-dimensional Lorentzian manifold that describes a **static black hole** in general relativity

https://nbviewer.org/github/egourgoulhon/SageMathTour/blob/master/Notebooks/demo_pseudo_Riemannian_Schwarzschild.ipynb

Outline

- 1 SageMath and its differential geometry capabilities
- 2 SageMath implementation of tensor fields
- 3 Example: Schwarzschild spacetime
- 4 Other examples**
- 5 Conclusions

Other examples

- **Computation of geodesics in Kerr spacetime** (rotating black hole):

[https:](https://nbviewer.jupyter.org/github/BlackHolePerturbationToolkit/kerrgeodesic_gw/blob/master/Notebooks/Kerr_geodesics.ipynb)

[//nbviewer.jupyter.org/github/BlackHolePerturbationToolkit/kerrgeodesic_gw/blob/master/Notebooks/Kerr_geodesics.ipynb](https://nbviewer.jupyter.org/github/BlackHolePerturbationToolkit/kerrgeodesic_gw/blob/master/Notebooks/Kerr_geodesics.ipynb)

- **The spheres \mathbb{S}^2 and \mathbb{S}^3 :**

https://nbviewer.org/github/sagemanifolds/SageManifolds/blob/master/Notebooks/SM_sphere_S2.ipynb

https://nbviewer.org/github/sagemanifolds/SageManifolds/blob/master/Notebooks/SM_sphere_S3_Hopf.ipynb

Image of an accretion disk surrounding a Schwarzschild BH

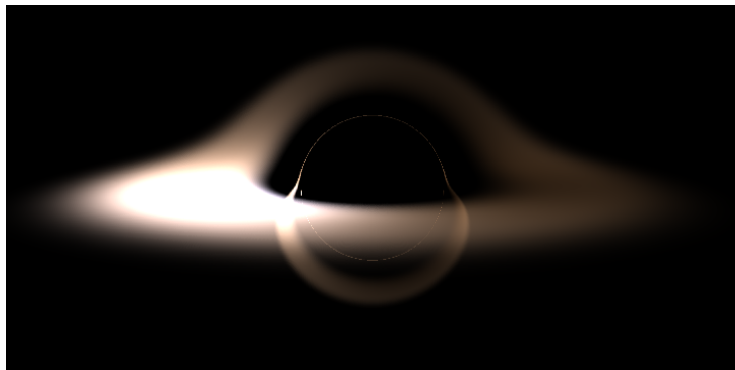


Image computed with SageMath by integrating null geodesics, cf. the notebook https://nbviewer.jupyter.org/github/sagemanifolds/SageManifolds/blob/master/Notebooks/SM_black_hole_rendering.ipynb

Outline

- 1 SageMath and its differential geometry capabilities
- 2 SageMath implementation of tensor fields
- 3 Example: Schwarzschild spacetime
- 4 Other examples
- 5 Conclusions**

Conclusions

Symbolic calculus on manifolds in the free Python-based system **SageMath**

- runs on fully specified smooth manifolds (described by an atlas)

Conclusions

Symbolic calculus on manifolds in the free Python-based system **SageMath**

- runs on fully specified smooth manifolds (described by an atlas)
- is not limited to a single coordinate chart or vector frame

Conclusions

Symbolic calculus on manifolds in the free Python-based system **SageMath**

- runs on fully specified smooth manifolds (described by an atlas)
- is not limited to a single coordinate chart or vector frame
- runs on parallelizable and non-parallelizable manifolds

Conclusions

Symbolic calculus on manifolds in the free Python-based system **SageMath**

- runs on fully specified smooth manifolds (described by an atlas)
- is not limited to a single coordinate chart or vector frame
- runs on parallelizable and non-parallelizable manifolds
- is independent of the symbolic engine (e.g. *Pynac/Maxima*, *SymPy*,...) used to perform calculus at the level of coordinate expressions

Conclusions

Symbolic calculus on manifolds in the free Python-based system **SageMath**

- runs on fully specified smooth manifolds (described by an atlas)
- is not limited to a single coordinate chart or vector frame
- runs on parallelizable and non-parallelizable manifolds
- is independent of the symbolic engine (e.g. *Pynac/Maxima*, *SymPy*,...) used to perform calculus at the level of coordinate expressions

Conclusions

Symbolic calculus on manifolds in the free Python-based system **SageMath**

- runs on fully specified smooth manifolds (described by an atlas)
- is not limited to a single coordinate chart or vector frame
- runs on parallelizable and non-parallelizable manifolds
- is independent of the symbolic engine (e.g. *Pynac/Maxima*, *SymPy*,...) used to perform calculus at the level of coordinate expressions

Many examples available at

<https://sagemanifolds.obspm.fr/examples.html>

Want to join the SageManifolds project or to simply stay tuned?

visit <https://sagemanifolds.obspm.fr/>
(download, documentation, example notebooks, mailing list)