Tensors DSLs and Curved Space-Time

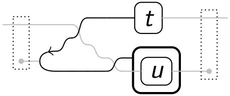
P Jansson[1] J-P Bernardy[2]

[1]Chalmers[2]UGOT

2024-03-01

Tensor calculus uses a terse but effective language for expressing physical laws. It was instrumental already a century ago in the formulation of Einstein's general relativity. We have implemented two tensor DSLs embedded in Haskell, the conversions between them, and helper functions to view the tensors in Einstein index notation and as diagrams. As an example we express Einstein's General Relativity equation for the curvature of space-time and verify that the Schwarzschild metric tensor is a solution. [based on arXiv.org/abs/2312.02664]

Contributions

- We provide two tensor EDSLs connected by automatic conversions: the index-based ALBERT and the morphism-based ROGER.
- The index-notation, diagram notation and matrices can be generated.
- example t u = contract (λ ${}^{\mathrm{i}}{}_{\mathrm{i}}$ $\rightarrow$ t ${}^{\mathrm{i}}$ $\star$ deriv ${}_{\mathrm{i}}$ u) can either:
  1. render itself as the index-notation expression $t^i \nabla_i u$;
  2. render itself as the diagram  or
  3. run on matrix representations of the tensors t and u and compute the result
- Motivating example: Einstein's General Relativity equation for the curvature of space-time and the Schwarzschild metric tensor.

Tensor definitions from introductory texts

> *A tensor of rank n is an array of 3^n values (in 3-D space) called "tensor components" that combine with multiple directional indicators (basis vectors) to form a quantity that does not vary as the coordinate system is changed.* (Fleisch 2011)

> *An nth-rank tensor in m-dimensional space is a mathematical object that has n indices and m^n components and obeys certain transformation rules.*
> (Rowland and Weisstein 2023)

These are typical - but rather confusing: there is this underlying "mathematical object" but also "3^n ... components" which "obeys certain transformation rules".
The components are described as "values", but as they "transform" they are more properly thought of as functions (but from what?).

Vector space

A vector space is a commutative group v equipped with a compatible notion of scaling.

$$
\begin{array}{l}
\textbf{class}\ \mathrm{Group}\ \mathrm{v}\ \textbf{where} \\
\quad \begin{array}{ll}
(+) & :: \mathrm{v} \to \mathrm{v} \to \mathrm{v} \\
0 & :: \mathrm{v} \\
\mathrm{negate} & :: \mathrm{v} \to \mathrm{v}
\end{array}
\end{array}
$$

$$
\begin{array}{l}
\textbf{class}\ (\mathrm{Group}\ \mathrm{v}) \Rightarrow \mathrm{VectorSpace}\ \mathrm{v}\ \textbf{where} \\
\quad (\triangleleft) :: \mathrm{S} \to \mathrm{v} \to \mathrm{v}
\end{array}
$$

Vector space

A vector space is a commutative group v equipped with a compatible notion of scaling.

$$
\begin{array}{l}
\textbf{class}\ \mathsf{Group}\ \mathsf{v}\ \textbf{where} \\
\quad (+) \quad :: \mathsf{v} \to \mathsf{v} \to \mathsf{v} \\
\quad 0 \qquad :: \mathsf{v} \\
\quad \mathsf{negate} :: \mathsf{v} \to \mathsf{v}
\end{array}
\qquad
\begin{array}{l}
\textbf{class}\ (\mathsf{Group}\ \mathsf{v}) \Rightarrow \mathsf{VectorSpace}\ \mathsf{v}\ \textbf{where} \\
\quad (\triangleleft) :: \mathsf{S} \to \mathsf{v} \to \mathsf{v}
\end{array}
$$

Example space:

Start from $S = \mathbb{R}$ and a 2D vector space V with basis vectors e_1 and e_2.

Any vector $\mathsf{v} : \mathsf{V}$ can be written as $\mathsf{v} = \mathsf{v}^1 \triangleleft \mathsf{e}_1 + \mathsf{v}^2 \triangleleft \mathsf{e}_2$.

Category of vector spaces

Consider a category of (finite-dimensional) vector spaces (objects) and linear transformations between them (arrows/morphisms). These arrows are the tensors.

Category of vector spaces

Consider a category of (finite-dimensional) vector spaces (objects) and linear transformations between them (arrows/morphisms). These arrows are the tensors.

$$
\begin{aligned}
&\textbf{class}\ \mathrm{Category}\ z\ \textbf{where}\\
&\quad \mathrm{id}\ \ :: \mathrm{a} \overset{z}{\rightsquigarrow} \mathrm{a}\\
&\quad (\circ) :: (\mathrm{b} \overset{z}{\rightsquigarrow} \mathrm{c}) \to (\mathrm{a} \overset{z}{\rightsquigarrow} \mathrm{b}) \to (\mathrm{a} \overset{z}{\rightsquigarrow} \mathrm{c})
\end{aligned}
$$

Category of vector spaces

Consider a category of (finite-dimensional) vector spaces (objects) and linear transformations between them (arrows/morphisms). These arrows are the tensors.

$$
\begin{array}{l}
\textbf{class}\ \mathrm{Category}\ z\ \textbf{where} \\
\quad \mathrm{id}\ \ :: \mathrm{a} \overset{z}{\rightsquigarrow} \mathrm{a} \\
\quad (\circ) :: (\mathrm{b} \overset{z}{\rightsquigarrow} \mathrm{c}) \rightarrow (\mathrm{a} \overset{z}{\rightsquigarrow} \mathrm{b}) \rightarrow (\mathrm{a} \overset{z}{\rightsquigarrow} \mathrm{c})
\end{array}
$$

$i - j$	$i - \boxed{t} - \boxed{u} - j$
id	$\mathrm{u} \circ \mathrm{t}$
$\delta_i{}^j$	$u_k{}^j t_i{}^k$

Category of vector spaces

Consider a category of (finite-dimensional) vector spaces (objects) and linear transformations between them (arrows/morphisms). These arrows are the tensors.

class Category z **where**
  id :: a $\overset{z}{\leadsto}$ a
  (∘) :: (b $\overset{z}{\leadsto}$ c) → (a $\overset{z}{\leadsto}$ b) → (a $\overset{z}{\leadsto}$ c)

$i - j$	$i - \boxed{t} - \boxed{u} - j$
id	u ∘ t
$\delta_i{}^j$	$u_k{}^j t_i{}^k$

- The unit type represents the (trivial) 1-Dim vector space of the real numbers. The canonical basis element is 1 and any value in this vector space can be written s ◃ 1.
- Example object: The 2D example space V.
- Example arrow: the linear transformation rot_{90} :: V $\overset{z}{\leadsto}$ V. In the orthonormal basis it can be represented as the matrix $\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$

Linear transformations / morphisms

Linear maps from v to w themselves form a vector space.

$$
\begin{array}{l}
\textbf{instance}\ (\mathrm{Group}\ w) \Rightarrow \mathrm{Group}\ (v \longrightarrow w)\ \textbf{where}\\
\quad \mathrm{negate}\ f = \lambda\, v \to \mathrm{negate}\ (f\ v)\\
\quad f + g = \lambda\, v \to f\ v + g\ v\\
\quad 0 \qquad\ = \lambda\, v \to 0\\
\textbf{instance}\ (\mathrm{VectorSpace}\ v, \mathrm{VectorSpace}\ w) \Rightarrow \mathrm{VectorSpace}\ (v \longrightarrow w)\ \textbf{where}\\
\quad c \triangleleft f = \lambda\, v \to c \triangleleft f\ v
\end{array}
$$

Linear transformations / morphisms

Linear maps from v to w themselves form a vector space.

$$
\begin{array}{l}
\textbf{instance}\ (\mathrm{Group}\ w) \Rightarrow \mathrm{Group}\ (v \longrightarrow w)\ \textbf{where} \\
\quad \mathrm{negate}\ f = \lambda\, v \rightarrow \mathrm{negate}\ (f\ v) \\
\quad f + g = \lambda\, v \rightarrow f\ v + g\ v \\
\quad 0 \qquad = \lambda\, v \rightarrow 0 \\
\textbf{instance}\ (\mathrm{VectorSpace}\ v, \mathrm{VectorSpace}\ w) \Rightarrow \mathrm{VectorSpace}\ (v \longrightarrow w)\ \textbf{where} \\
\quad c \triangleleft f = \lambda\, v \rightarrow c \triangleleft f\ v
\end{array}
$$

Examples:

$s : \mathbf{1} \overset{z}{\rightsquigarrow} \mathbf{1}$	determined by a scalar
$v : \mathbf{1} \overset{z}{\rightsquigarrow} V$	determined by a vector in V (two components)
$u : V \overset{z}{\rightsquigarrow} \mathbf{1}$	a "vector-eater" - determined by two components
$f : V \overset{z}{\rightsquigarrow} V$	a proper linear transformation - determined by a matrix

These examples are tensors of order (0,0), (0,1), (1,0), and (1,1).

Tensor order, rank, and product

- A tensor $t : V^m \overset{z}{\rightsquigarrow} V^n$ of order (m,n) informally
  "takes m vectors as inputs and produces n vectors as outputs".
- but the repeated product is *tensor product*, not cartesian product.
- The basis for a tensor product V $\otimes$ W is the cartesian product of the two bases.

Tensor order, rank, and product

- A tensor $t : V^m \overset{z}{\rightsquigarrow} V^n$ of order (m,n) informally "takes m vectors as inputs and produces n vectors as outputs".
- but the repeated product is *tensor product*, not cartesian product.
- The basis for a tensor product $\mathrm{V} \otimes \mathrm{W}$ is the cartesian product of the two bases.
- Given $\mathrm{v} : \mathrm{V}$ and $\mathrm{w} : \mathrm{W}$ their tensor product is $\mathrm{v} \otimes \mathrm{w} : \mathrm{V} \otimes \mathrm{W}$ (a "simple" tensor).
- In general a tensor is a sum of simple tensors.

Tensor order, rank, and product

- A tensor $t : V^m \overset{z}{\rightsquigarrow} V^n$ of order (m,n) informally
  "takes m vectors as inputs and produces n vectors as outputs".
- but the repeated product is *tensor product*, not cartesian product.
- The basis for a tensor product $\mathrm{V} \otimes \mathrm{W}$ is the cartesian product of the two bases.
- Given $\mathrm{v} : \mathrm{V}$ and $\mathrm{w} : \mathrm{W}$ their tensor product is $\mathrm{v} \otimes \mathrm{w} : \mathrm{V} \otimes \mathrm{W}$ (a "simple" tensor).
- In general a tensor is a sum of simple tensors.
- The *rank* of a tensor is the minimum number of simple tensors that sum to it.
- The rank of δ (in 2D-space) is 2, the rank of rot_{90} is 2, etc.

Dual space / basis

- A tensor $\mathrm{t} : \mathrm{V} \overset{z}{\rightsquigarrow} \mathbf{1}$ is called covector.
- We use the type V^* ("dual of V") for covectors.
- Given a basis e for V we can define a basis d for V^* such that $d_i(e_j) = \delta_i^j$.
- Then every covector is determined by n (covariant) components.

Dual space / basis

- A tensor $t : V \overset{z}{\rightsquigarrow} \mathbf{1}$ is called covector.
- We use the type V^* ("dual of V") for covectors.
- Given a basis e for V we can define a basis d for V^* such that $d_i(e_j) = \delta_i^j$.
- Then every covector is determined by n (covariant) components.
- More tensor examples:
  - an "inner product" has type $V \otimes V \overset{z}{\rightsquigarrow} \mathbf{1}$, thus order (2,0)
  - a metric has type $\mathbf{1} \overset{z}{\rightsquigarrow} V \otimes V$, thus order (0,2)

Einstein notation examples

- x^i denotes (the components of) a vector (of order (0,1))
- y_j denotes (the components of) a covector (of order (1,0))
- t_i^j refers to (components of) a linear map of order (1,1).
- $x^i\, y_j$ also has order (1,1) (mult. of tensors adds their orders)

Einstein notation examples

- x^i denotes (the components of) a vector (of order (0,1))
- y_j denotes (the components of) a covector (of order (1,0))
- t_i^j refers to (components of) a linear map of order (1,1).
- $x^i\,y_j$ also has order (1,1) (mult. of tensors adds their orders)
- Einstein convention: within a term, a repeated (dummy) index is implicitly summed over (called contraction).
- Example: $\delta_i^i = n$ where n is the dimension of the underlying vector space.

Einstein notation examples

- x^i denotes (the components of) a vector (of order (0,1))
- y_j denotes (the components of) a covector (of order (1,0))
- t_i^j refers to (components of) a linear map of order (1,1).
- $x^i\, y_j$ also has order (1,1) (mult. of tensors adds their orders)
- Einstein convention: within a term, a repeated (dummy) index is implicitly summed over (called contraction).
- Example: $\delta_i^i = n$ where n is the dimension of the underlying vector space.
- ${t_l}^j {u_m}^k {v_i}^{lm} + {v_i}^{jk}$ denotes a tensor of order (1,2).
  Its live indices are $_i$, ^{j}, ^{k}, and the indices m and l are dummies.

Implementation sketch

- We implement two DSLs: one for the morphisms and one for "index notation".
- The DSL for morphisms is called ROGER (after Penrose) and is based on Symmetric Monoidal Categories (SMCs).
- The DSL for index notation is called ALBERT (after Einstein) and is based on (type-)linear functions between "ports" .
- The DSLs are connected through these isomorphisms:
  $\text{tensorEmbed} :: (\text{CompactClosed}\ z) \Rightarrow \quad (\mathrm{a} \overset{z}{\rightsquigarrow} \mathrm{b}) \to (\forall\, \mathrm{r}\,.\ \mathrm{P}\ z\ \mathrm{r}\ \mathrm{a} \multimap \mathrm{P}\ z\ \mathrm{r}\ \mathrm{b}^* \multimap \mathrm{R}\ z\ \mathrm{r})$
  $\text{tensorEval} :: (\text{CompactClosed}\ z, \text{Additive}\ z) \Rightarrow (\forall\, \mathrm{r}\,.\ \mathrm{P}\ z\ \mathrm{r}\ \mathrm{a} \multimap \mathrm{P}\ z\ \mathrm{r}\ \mathrm{b}^* \multimap \mathrm{R}\ z\ \mathrm{r}) \to (\mathrm{a} \overset{z}{\rightsquigarrow} \mathrm{b})$
- tensorEmbed from ROGER to ALBERT
- tensorEval from ALBERT back to ROGER

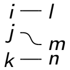
Symmetric monoidal categories (the morphism DSL Roger)

$$
\begin{array}{ll}
\multicolumn{2}{l}{\textbf{class}\ \mathsf{Category}\ z \Rightarrow \mathsf{SymmetricMonoidal}\ z\ \textbf{where}} \\
\quad (\otimes) & :: (a \overset{z}{\rightsquigarrow} b) \to (c \overset{z}{\rightsquigarrow} d) \to (a \otimes c) \overset{z}{\rightsquigarrow} (b \otimes d) \\
\quad \sigma & :: (a \otimes b) \overset{z}{\rightsquigarrow} (b \otimes a) \\
\quad \alpha & :: ((a \otimes b) \otimes c) \overset{z}{\rightsquigarrow} (a \otimes (b \otimes c)) \\
\quad \bar{\alpha} & :: (a \otimes (b \otimes c)) \overset{z}{\rightsquigarrow} ((a \otimes b) \otimes c) \\
\quad \rho & :: a \overset{z}{\rightsquigarrow} (a \otimes \mathbf{1}) \\
\quad \bar{\rho} & :: (a \otimes \mathbf{1}) \overset{z}{\rightsquigarrow} a
\end{array}
$$

Symmetric monoidal categories (the morphism DSL Roger)

class Category $z \Rightarrow$ SymmetricMonoidal z **where**

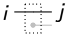
$$
\begin{array}{lll}
(\otimes) & :: & (\mathsf{a} \overset{z}{\rightsquigarrow} \mathsf{b}) \to (\mathsf{c} \overset{z}{\rightsquigarrow} \mathsf{d}) \to (\mathsf{a} \otimes \mathsf{c}) \overset{z}{\rightsquigarrow} (\mathsf{b} \otimes \mathsf{d}) \\
\sigma & :: & (\mathsf{a} \otimes \mathsf{b}) \overset{z}{\rightsquigarrow} (\mathsf{b} \otimes \mathsf{a}) \\
\alpha & :: & ((\mathsf{a} \otimes \mathsf{b}) \otimes \mathsf{c}) \overset{z}{\rightsquigarrow} (\mathsf{a} \otimes (\mathsf{b} \otimes \mathsf{c})) \\
\bar{\alpha} & :: & (\mathsf{a} \otimes (\mathsf{b} \otimes \mathsf{c})) \overset{z}{\rightsquigarrow} ((\mathsf{a} \otimes \mathsf{b}) \otimes \mathsf{c}) \\
\rho & :: & \mathsf{a} \overset{z}{\rightsquigarrow} (\mathsf{a} \otimes \mathbf{1}) \\
\bar{\rho} & :: & (\mathsf{a} \otimes \mathbf{1}) \overset{z}{\rightsquigarrow} \mathsf{a}
\end{array}
$$



α

$\delta_i{}^l \delta_j{}^m \delta_k{}^n$

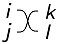
$(\mathsf{T} \otimes \mathsf{T}) \otimes \mathsf{T} \overset{z}{\rightsquigarrow} \mathsf{T} \otimes (\mathsf{T} \otimes \mathsf{T})$



ρ

$\delta_i{}^j$

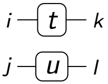
$\mathsf{T} \overset{z}{\rightsquigarrow} \mathsf{T} \otimes \mathbf{1}$



σ

$\delta_i{}^l \delta_j{}^k$

$\mathsf{T} \otimes \mathsf{T} \overset{z}{\rightsquigarrow} \mathsf{T} \otimes \mathsf{T}$



$\mathsf{t} \otimes \mathsf{u}$

$t_i{}^k u_j{}^l$

$\mathsf{T} \otimes \mathsf{T} \overset{z}{\rightsquigarrow} \mathsf{T} \otimes \mathsf{T}$

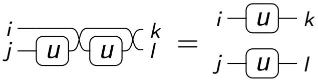
Figure: Diagram, categorical, and index notations for morphisms of SMCs (of example types).

Tensor algebra / string diagram examples

Assume an abstract tensor $u : V \overset{z}{\rightsquigarrow} V$

- Is $\sigma \circ (\mathrm{id} \otimes u) \circ \sigma \circ (\mathrm{id} \otimes u)$ equivalent to $u \otimes u$?

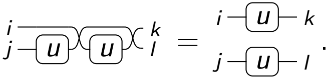
Assume an abstract tensor $\mathrm{u} : \mathrm{V} \overset{z}{\rightsquigarrow} \mathrm{V}$

- Is $\sigma \circ (\mathrm{id} \otimes \mathrm{u}) \circ \sigma \circ (\mathrm{id} \otimes \mathrm{u})$ equivalent to $\mathrm{u} \otimes \mathrm{u}$?
- In diagram form: 

.

Tensor algebra / string diagram examples

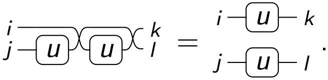
Assume an abstract tensor $\mathrm{u} : \mathrm{V} \overset{z}{\rightsquigarrow} \mathrm{V}$

- Is $\sigma \circ (\mathrm{id} \otimes \mathrm{u}) \circ \sigma \circ (\mathrm{id} \otimes \mathrm{u})$ equivalent to $\mathrm{u} \otimes \mathrm{u}$?
- In diagram form: 

- We can just "pull at the wires" to see that they are equal.

Tensor algebra / string diagram examples

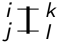
Assume an abstract tensor $\mathrm{u} : \mathrm{V} \overset{z}{\rightsquigarrow} \mathrm{V}$

- Is $\sigma \circ (\mathrm{id} \otimes \mathrm{u}) \circ \sigma \circ (\mathrm{id} \otimes \mathrm{u})$ equivalent to $\mathrm{u} \otimes \mathrm{u}$?
- In diagram form: 

- We can just "pull at the wires" to see that they are equal.

Another example (of order (2,2)):

- $\mathrm{antisym} = \mathrm{id} - \sigma :: \mathrm{T} \otimes \mathrm{T} \overset{z}{\rightsquigarrow} \mathrm{T} \otimes \mathrm{T}$
- Diagram: 

- Special notation: 

[Optional] Compact Closed Category: connecting a space with its dual

In CompactClosed, every object has a dual (generalisation of co-vector space).

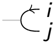
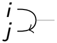
$$\textbf{class}\ (\mathsf{SymmetricMonoidal}\ z) \Rightarrow \mathsf{CompactClosed}\ z\ \textbf{where}$$
$$\quad \eta :: \mathbf{1} \overset{z}{\leadsto} (\mathsf{a}^* \otimes \mathsf{a})$$
$$\quad \epsilon :: (\mathsf{a} \otimes \mathsf{a}^*) \overset{z}{\leadsto} \mathbf{1}$$

[Optional] Compact Closed Category: connecting a space with its dual

In CompactClosed, every object has a dual (generalisation of co-vector space).

$$\textbf{class}\ (\mathsf{SymmetricMonoidal}\ z) \Rightarrow \mathsf{CompactClosed}\ z\ \textbf{where}$$
$$\eta :: \mathbf{1} \overset{z}{\rightsquigarrow} (a^* \otimes a)$$
$$\epsilon :: (a \otimes a^*) \overset{z}{\rightsquigarrow} \mathbf{1}$$

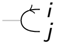
Unit and counit:

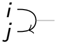
	
η	ϵ
${\delta_j}^i$	${\delta_i}^j$
$\mathbf{1} \overset{z}{\rightsquigarrow} T^* \otimes T$	$T \otimes T^* \overset{z}{\rightsquigarrow} \mathbf{1}$

[Optional] Compact Closed Category: connecting a space with its dual

In CompactClosed, every object has a dual (generalisation of co-vector space).

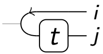
$$\textbf{class}\ (\mathsf{SymmetricMonoidal}\ z) \Rightarrow \mathsf{CompactClosed}\ z\ \textbf{where}$$
$$\eta :: \mathbf{1} \overset{z}{\rightsquigarrow} (\mathsf{a}^* \otimes \mathsf{a})$$
$$\epsilon :: (\mathsf{a} \otimes \mathsf{a}^*) \overset{z}{\rightsquigarrow} \mathbf{1}$$

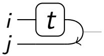


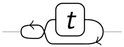


Unit and counit:

η	ϵ
${\delta_j}^i$	${\delta_i}^j$
$\mathbf{1} \overset{z}{\rightsquigarrow} \mathsf{T}^* \otimes \mathsf{T}$	$\mathsf{T} \otimes \mathsf{T}^* \overset{z}{\rightsquigarrow} \mathbf{1}$







Examples:

$(\mathsf{id} \otimes \mathsf{t}) \circ \eta$	$\epsilon \circ (\mathsf{t} \otimes \mathsf{id})$	$\epsilon \circ (\mathsf{t} \otimes \mathsf{id}) \circ \sigma \circ \eta$
${t_i}^j$	${t_i}^j$	${t_i}^i$
$\mathbf{1} \overset{z}{\rightsquigarrow} \mathsf{T}^* \otimes \mathsf{T}$	$\mathsf{T} \otimes \mathsf{T}^* \overset{z}{\rightsquigarrow} \mathbf{1}$	$\mathbf{1} \overset{z}{\rightsquigarrow} \mathbf{1}$

[Optional] Matrix Instances

newtype M_e a b = Tab (a → b → S)

[Optional] Matrix Instances

$$
\begin{array}{l}
\textbf{newtype}\ M_e\ \mathsf{a}\ \mathsf{b} = \mathsf{Tab}\ (\mathsf{a} \to \mathsf{b} \to \mathsf{S}) \\
\textbf{instance}\ \mathsf{Category}\ M_e\ \textbf{where} \\
\quad \mathsf{id} = \mathsf{Tab}\ \delta \\
\quad \mathsf{Tab}\ \mathsf{g} \circ \mathsf{Tab}\ \mathsf{f} = \mathsf{Tab}\ (\lambda\ \mathsf{i}\ \mathsf{j} \to \mathsf{summation}\ (\lambda\ \mathsf{k} \to \mathsf{f}\ \mathsf{i}\ \mathsf{k} * \mathsf{g}\ \mathsf{k}\ \mathsf{j})) \\
\delta\ \mathsf{i}\ \mathsf{j} = \textbf{if}\ \mathsf{i} == \mathsf{j}\ \textbf{then}\ 1\ \textbf{else}\ 0
\end{array}
$$

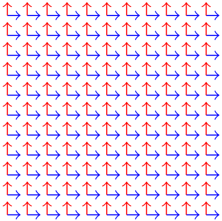
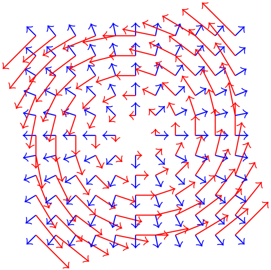
[Optional] Matrix Instances

$$
\begin{array}{l}
\textbf{newtype}\ M_e\ a\ b = \mathrm{Tab}\ (a \to b \to S) \\
\textbf{instance}\ \mathrm{Category}\ M_e\ \textbf{where} \\
\quad \mathrm{id} = \mathrm{Tab}\ \delta \\
\quad \mathrm{Tab}\ g \circ \mathrm{Tab}\ f = \mathrm{Tab}\ (\lambda\, i\, j \to \mathrm{summation}\ (\lambda\, k \to f\ i\ k * g\ k\ j)) \\
\delta\ i\ j = \textbf{if}\ i == j\ \textbf{then}\ 1\ \textbf{else}\ 0 \\
\textbf{instance}\ \mathrm{SymmetricMonoidal}\ M_e\ \textbf{where} \\
\quad (\otimes) = \mathrm{kroneckerProduct} \\
\quad \rho = \mathrm{Tab}\ (\lambda\, x\ (y, ()) \to \delta\ x\ y) \\
\quad \bar{\rho} = \mathrm{Tab}\ (\lambda\, (y, ())\ x \to \delta\ x\ y) \\
\quad \alpha = \mathrm{Tab}\ (\lambda\, ((x, y), z)\ (x', (y', z')) \to \delta\ ((x, y), z)\ ((x', y'), z')) \\
\quad \bar{\alpha} = \mathrm{Tab}\ (\lambda\, (x', (y', z'))\ ((x, y), z) \to \delta\ ((x, y), z)\ ((x', y'), z')) \\
\quad \sigma = \mathrm{Tab}\ (\lambda\, (x, y)\ (y', x') \to \delta\ (x, y)\ (x', y')) \\
\mathrm{kroneckerProduct}\ (\mathrm{Tab}\ f)\ (\mathrm{Tab}\ g) = \mathrm{Tab}\ (\lambda\, (i, k)\ (j, l) \to f\ i\ j * g\ k\ l)
\end{array}
$$

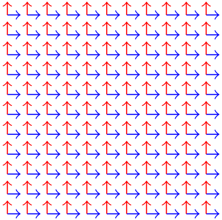
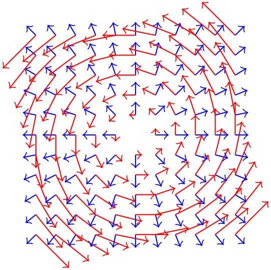
[Optional] Summing up the core of the index EDSL interface

$$
\begin{aligned}
\mathsf{tensorEmbed} &:: (\mathsf{CompactClosed}\ z) \Rightarrow \\
&\quad (\mathsf{a} \overset{z}{\rightsquigarrow} \mathsf{b}) \rightarrow (\forall\, \mathsf{r}.\ \mathsf{P}\ z\ \mathsf{r}\ \mathsf{a} \multimap \mathsf{P}\ z\ \mathsf{r}\ \mathsf{b}^{*} \multimap \mathsf{R}\ z\ \mathsf{r}) \\
\mathsf{tensorEval} &:: (\mathsf{CompactClosed}\ z, \mathsf{Additive}\ z) \Rightarrow \\
&\quad (\forall\, \mathsf{r}.\ \mathsf{P}\ z\ \mathsf{r}\ \mathsf{a} \multimap \mathsf{P}\ z\ \mathsf{r}\ \mathsf{b}^{*} \multimap \mathsf{R}\ z\ \mathsf{r}) \rightarrow (\mathsf{a} \overset{z}{\rightsquigarrow} \mathsf{b}) \\
\mathsf{plus} &:: (\mathsf{Bool} \rightarrow \mathsf{R}\ z\ \mathsf{r}) \multimap \mathsf{R}\ z\ \mathsf{r} \\
(\star) &:: \mathsf{SymmetricMonoidal}\ z \Rightarrow \mathsf{R}\ z\ \mathsf{r} \multimap \mathsf{R}\ z\ \mathsf{r} \multimap \mathsf{R}\ z\ \mathsf{r} \\
\mathsf{zeroTensor} &:: (\mathsf{CompactClosed}\ z, \mathsf{Additive}\ z) \Rightarrow \mathsf{P}\ z\ \mathsf{r}\ \mathsf{a} \multimap \mathsf{R}\ z\ \mathsf{r} \\
\mathsf{constant} &:: (\mathsf{SymmetricMonoidal}\ z, \mathsf{VectorSpace}\ (\mathbf{1} \overset{z}{\rightsquigarrow} \mathbf{1})) \Rightarrow \mathsf{S} \rightarrow \mathsf{R}\ z\ \mathsf{r} \\
\mathsf{delta} &:: \mathsf{CompactClosed}\ z \Rightarrow \mathsf{P}\ z\ \mathsf{r}\ \mathsf{a} \multimap \mathsf{P}\ z\ \mathsf{r}\ \mathsf{a}^{*} \multimap \mathsf{R}\ z\ \mathsf{r} \\
\mathsf{contract} &:: \mathsf{CompactClosed}\ z \Rightarrow (\mathsf{P}\ z\ \mathsf{r}\ \mathsf{a}^{*} \multimap \mathsf{P}\ z\ \mathsf{r}\ \mathsf{a} \multimap \mathsf{R}\ z\ \mathsf{r}) \multimap \mathsf{R}\ z\ \mathsf{r}
\end{aligned}
$$

Tensor fields

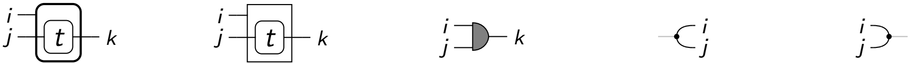
- Moving from tensor algebra to tensor calculus
- We now have a vector at every position (of some manifold)...
- ... and even the basis can change with position.
- Example: Cartesian: , polar: 

Tensor fields

- Moving from tensor algebra to tensor calculus
- We now have a vector at every position (of some manifold)...
- ... and even the basis can change with position.
- Example: Cartesian: , polar: 
- Spatial derivatives: the derivative of a scalar field s is a covector field: its gradient.
- The derivative of a tensor increases the (covariant) order of its argument.

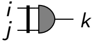
  $\nabla :: (a \overset{z}{\rightsquigarrow} b) \to ((v \otimes a) \overset{z}{\rightsquigarrow} b)$

Derivatives, Christoffel symbols, and metrics



$\nabla\,\mathrm{t}$	$\partial\,\mathrm{t}$	Γ	g'	g
$\nabla_i {t_j}^k$	$\partial_i {t_j}^k$	${\Gamma_{ij}}^k$	g'^{ij}	g_{ij}
$T_z \otimes T_z \overset{z}{\rightsquigarrow} T_z$	$T_z \otimes T_z \overset{z}{\rightsquigarrow} T_z$	$T_z \otimes T_z \overset{z}{\rightsquigarrow} T_z$	$\mathbf{1} \overset{z}{\rightsquigarrow} T_z \otimes T_z$	$T_z \otimes T_z \overset{z}{\rightsquigarrow} \mathbf{1}$

Figure: Tensor field primitives in various notations, and their types.

The variation of the basis is measured by the *Christoffel symbol*, denoted Γ.
Different choices of local basis field for the *same* manifold will yield different values for it.
The Christoffel symbol is symmetric in the lower indices:  $= 0$.

Covariant derivative and affinity

In ROGER we can express the general covariant derivative as follows

$$\nabla\, \mathsf{t} = \partial\, \mathsf{t} - (\mathsf{t} \circ \mathsf{affinity}) + (\mathsf{affinity} \circ (\mathsf{id} \otimes \mathsf{t})) \tag{1}$$

The complexity is pushed down into affinity, which is invoked "on both sides" of t.
Some simple cases (in Einstein notation) are:

$$\begin{aligned}
\nabla_i T &= \partial_i T \\
\nabla_i T_j &= \partial_i T_j - \Gamma^k_{ij} T_k \\
\nabla_i T^j &= \partial_i T^j + \Gamma^j_{ik} T^k \\
\nabla_i T^k_j &= \partial_i T^k_j - \Gamma^k_{il} T^l_j + \Gamma^l_{ij} T^k_l
\end{aligned}$$

[Optional] Derivative classes

Metric and co-metric:

$$
\begin{array}{l}
\textbf{class}\ (\mathrm{CompactClosed}\ z) \Rightarrow \mathrm{MetricCategory}\ z\ \textbf{where} \\
\quad \textbf{type}\ T_z \\
\quad g\ :: (T_z \otimes T_z) \overset{z}{\rightsquigarrow} \mathbf{1} \\
\quad g' :: \mathbf{1} \overset{z}{\rightsquigarrow} (T_z \otimes T_z)
\end{array}
$$

Covariant derivative (coordinate independent):

$$
\begin{array}{l}
\textbf{class}\ \mathrm{MetricCategory}\ z \Rightarrow \mathrm{ConnectionCategory}\ z\ \textbf{where} \\
\quad \nabla :: (\mathrm{a} \overset{z}{\rightsquigarrow} \mathrm{b}) \to ((T_z \otimes \mathrm{a}) \overset{z}{\rightsquigarrow} \mathrm{b})
\end{array}
$$

Partial derivative (coordinate dependent):

$$
\begin{array}{l}
\textbf{class}\ \mathrm{MetricCategory}\ z \Rightarrow \mathrm{CoordinateCategory}\ z\ \textbf{where} \\
\quad \partial :: (\mathrm{a} \overset{z}{\rightsquigarrow} \mathrm{b}) \to ((T_z \otimes \mathrm{a}) \overset{z}{\rightsquigarrow} \mathrm{b})
\end{array}
$$

Christoffel symbol:

$$
\Gamma_{ij}{}^{k} = \frac{1}{2} g'^{lk} \partial_j g_{il} + \frac{1}{2} g'^{mk} \partial_i g_{jm} - \frac{1}{2} g'^{nk} \partial_n g_{ji} \tag{2}
$$

Derivative laws (in Roger)

The product law is expressed as two cases, one for each of the $(\circ)$ and $(\otimes)$ operators:

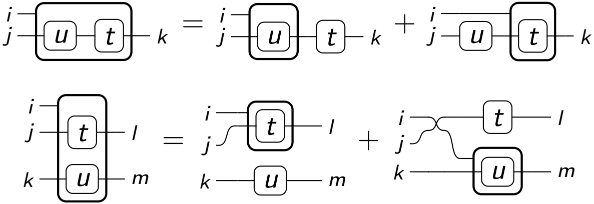
$$\nabla\, \mathsf{t} \circ \mathsf{u} = \mathsf{t} \circ (\nabla\, \mathsf{u}) + (\nabla\, \mathsf{t}) \circ (\mathsf{id} \otimes \mathsf{u})$$
$$\nabla\, (\mathsf{t} \otimes \mathsf{u}) = (\nabla\, \mathsf{t} \otimes \mathsf{u}) \circ \bar{\alpha} + (\mathsf{t} \otimes \nabla\, \mathsf{u}) \circ \alpha \circ (\sigma \otimes \mathsf{id}) \circ \bar{\alpha}$$

Derivative laws (in ROGER)

The product law is expressed as two cases, one for each of the $(\circ)$ and $(\otimes)$ operators:

$$\nabla\, t \circ u = t \circ (\nabla\, u) + (\nabla\, t) \circ (\mathrm{id} \otimes u)$$
$$\nabla\, (t \otimes u) = (\nabla\, t \otimes u) \circ \bar{\alpha} + (t \otimes \nabla\, u) \circ \alpha \circ (\sigma \otimes \mathrm{id}) \circ \bar{\alpha}$$

The corresponding diagrams provide give a complementing view:



(3)

(4)

General Relativity and Curved Space-Time

General relativity can be summarised as "matter curves space-time".
This informal statement can be expressed as a tensor equation as follows

$$R^{k}{}_{kij} + \frac{1}{2} g_{ij} g'^{lm} R^{n}{}_{nlm} = \kappa T_{ij} \tag{5}$$

General Relativity and Curved Space-Time

General relativity can be summarised as "matter curves space-time".
This informal statement can be expressed as a tensor equation as follows

$$R^{k}{}_{kij} + \frac{1}{2} g_{ij} g'^{lm} R^{n}{}_{nlm} = \kappa T_{ij} \tag{5}$$

T_{ij} is the "energy-momentum tensor" (with mass as a special case)
κ is the gravitational constant.

General Relativity and Curved Space-Time

General relativity can be summarised as “matter curves space-time”.
This informal statement can be expressed as a tensor equation as follows

$$R^k{}_{kij} + \frac{1}{2} g_{ij} g'^{lm} R^n{}_{nlm} = \kappa T_{ij} \tag{5}$$

T_{ij} is the “energy-momentum tensor” (with mass as a special case)
κ is the gravitational constant.
$R^l{}_{ijk}$ is the “Riemann curvature tensor” (defined in terms of the metric g_{ij}).
The (implicit) variable to be solved for is the metric.

General Relativity and Curved Space-Time

General relativity can be summarised as "matter curves space-time".
This informal statement can be expressed as a tensor equation as follows

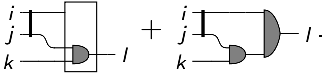
$$R^{k}{}_{kij} + \frac{1}{2} g_{ij} g'^{lm} R^{n}{}_{nlm} = \kappa T_{ij} \tag{5}$$

T_{ij} is the "energy-momentum tensor" (with mass as a special case)
κ is the gravitational constant.
$R^{l}{}_{ijk}$ is the "Riemann curvature tensor" (defined in terms of the metric g_{ij}).
The (implicit) variable to be solved for is the metric.
We will not solve it, but "just" check that the Schwarzschild metric is a solution.

Definition: Riemann curvature

The Riemann curvature is a 4-tensor, implemented by the following identity:
$R^{l}{}_{ijk} = \partial_i \Gamma_{jk}{}^{l} - \partial_j \Gamma_{ik}{}^{l} + \Gamma_{im}{}^{l}\Gamma_{jk}{}^{m} - \Gamma_{jn}{}^{l}\Gamma_{ik}{}^{n}$.
Each pair of terms is the antisymmetric part of a 4-tensor.

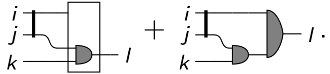
Thus we can make the diagram notation a sum of two terms:



Definition: Riemann curvature

The Riemann curvature is a 4-tensor, implemented by the following identity:
$R^l{}_{ijk} = \partial_i \Gamma_{jk}{}^l - \partial_j \Gamma_{ik}{}^l + \Gamma_{im}{}^l \Gamma_{jk}{}^m - \Gamma_{jn}{}^l \Gamma_{ik}{}^n$.
Each pair of terms is the antisymmetric part of a 4-tensor.

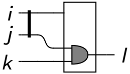
Thus we can make the diagram notation a sum of two terms:



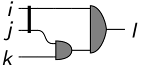
In ROGER: $(\partial\Gamma + \Gamma \circ (\mathrm{id} \otimes \Gamma)) \circ \alpha \circ ((\mathrm{id} - \sigma) \otimes \mathrm{id})$

Definition: Riemann curvature

The Riemann curvature is a 4-tensor, implemented by the following identity:
$R^{l}{}_{ijk} = \partial_i \Gamma_{jk}{}^{l} - \partial_j \Gamma_{ik}{}^{l} + \Gamma_{im}{}^{l}\Gamma_{jk}{}^{m} - \Gamma_{jn}{}^{l}\Gamma_{ik}{}^{n}$.
Each pair of terms is the antisymmetric part of a 4-tensor.
Thus we can make the diagram notation a sum of two terms:



$+$



.

In ROGER: $(\partial\,\Gamma + \Gamma \circ (\mathrm{id} \otimes \Gamma)) \circ \alpha \circ ((\mathrm{id} - \sigma) \otimes \mathrm{id})$
Note: that ∂ and Γ are coordinate-dependent — a proof is needed to show that the Riemann curvature is independent (and thus a tensor).

[Optional] Riemann curvature encoding in ALBERT

The above definition can be encoded directly in ALBERT as follows:

$$
\begin{array}{l}
\mathsf{curvature} :: \mathsf{CoordinateCategory}\; z \Rightarrow \mathsf{P}\; z\; \mathsf{r}\; {T_z}^{*} \multimap \mathsf{P}\; z\; \mathsf{r}\; T_z \multimap \mathsf{P}\; z\; \mathsf{r}\; T_z \multimap \mathsf{P}\; z\; \mathsf{r}\; T_z \multimap \mathsf{R}\; z\; \mathsf{r} \\
\mathsf{curvature}^{\,\mathsf{l}}{}_{\mathsf{k}\,\mathsf{i}\,\mathsf{j}} \\
\quad = \mathsf{plus}\; (\lambda\, \mathsf{a} \rightarrow \mathbf{case}\; \mathsf{a}\; \mathbf{of} \\
\qquad \mathsf{True} \;\rightarrow \mathsf{minus}\; (\lambda\, \mathsf{b} \rightarrow \mathbf{case}\; \mathsf{b}\; \mathbf{of} \\
\qquad\quad \mathsf{True} \;\rightarrow \mathsf{partial}_{\mathsf{i}}\; (\mathsf{christoffel}_{\mathsf{j}\,\mathsf{k}}{}^{\mathsf{l}}) \\
\qquad\quad \mathsf{False} \rightarrow \mathsf{partial}_{\mathsf{j}}\; (\mathsf{christoffel}_{\mathsf{i}\,\mathsf{k}}{}^{\mathsf{l}})) \\
\qquad \mathsf{False} \rightarrow \mathsf{contract}\; (\lambda\, {}^{\mathsf{m}}{}_{\mathsf{m}} \rightarrow \mathsf{minus}\; (\lambda\, \mathsf{b} \rightarrow \mathbf{case}\; \mathsf{b}\; \mathbf{of} \\
\qquad\quad \mathsf{True} \;\rightarrow \mathsf{christoffel}_{\mathsf{i}\,\mathsf{m}}{}^{\mathsf{l}} \star \mathsf{christoffel}_{\mathsf{j}\,\mathsf{k}}{}^{\mathsf{m}} \\
\qquad\quad \mathsf{False} \rightarrow \mathsf{christoffel}_{\mathsf{j}\,\mathsf{m}}{}^{\mathsf{l}} \star \mathsf{christoffel}_{\mathsf{i}\,\mathsf{k}}{}^{\mathsf{m}})))
\end{array}
$$

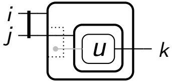
Even though it is defined in terms of Christoffel symbols, the Riemann curvature (as a geometric object) does not depend on the choice of coordinates. This is a consequence of the Ricci identity (next).

Theorem: Ricci identity: a specification of the Riemann tensor

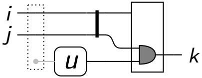
For every covector field u, $\nabla_i \nabla_j u^k - \nabla_j \nabla_i u^k = {R^k}_{ijl} u^l$
Note: This definition / specification only uses tensors, thus it follows that Riemann curvature is a tensor (does not depend on the choice of coordinates).

Theorem: Ricci identity: a specification of the Riemann tensor

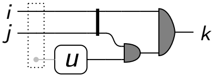
For every covector field u, $\nabla_i \nabla_j u^k - \nabla_j \nabla_i u^k = R^k{}_{ijl} u^l$
Note: This definition / specification only uses tensors, thus it follows that Riemann curvature is a tensor (does not depend on the choice of coordinates).
We carry out the proof using the diagram notation starting from the LHS:



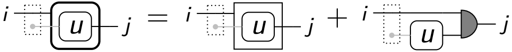
and aiming to reach the RHS:



+

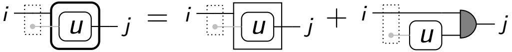


Proving the Ricci identity: helper lemma + first steps

Lemma: 

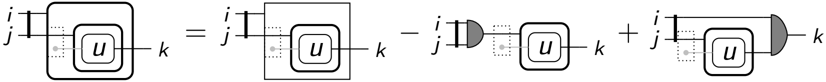
(an instance of the definition of the covariant derivative for a (0,1)-tensor)

Proving the Ricci identity: helper lemma + first steps

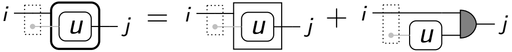
Lemma: 

(an instance of the definition of the covariant derivative for a (0,1)-tensor)

The LHS in the main theorem has the cov. derivative applied to a (1,1)-tensor: expand that into partial derivatives and affinity terms:

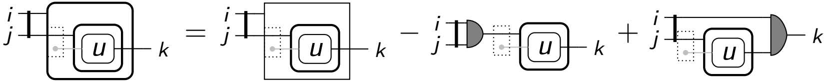


Proving the Ricci identity: helper lemma + first steps

Lemma: 

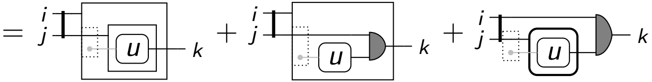
(an instance of the definition of the covariant derivative for a (0,1)-tensor)

The LHS in the main theorem has the cov. derivative applied to a (1,1)-tensor: expand that into partial derivatives and affinity terms:

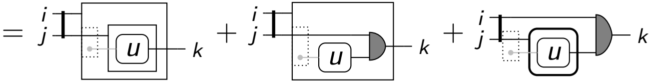


The middle term is zero, by symmetry of Γ.

Expanding the first term using the **Lemma** and linearity of partial derivatives, we get

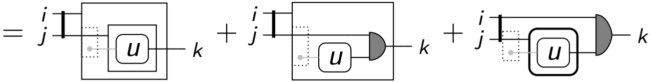


Proving the Ricci identity: cont.



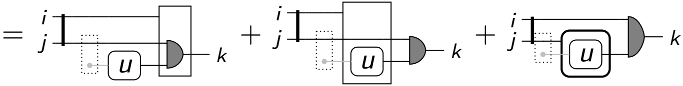
Because partial derivatives commute, the first term is zero.

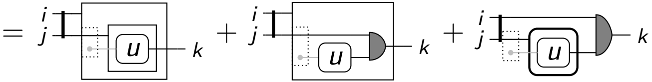
Proving the Ricci identity: cont.



Because partial derivatives commute, the first term is zero.

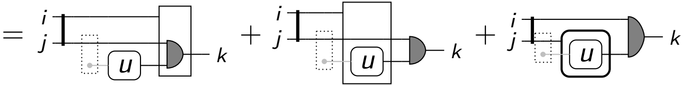
We expand the middle term using the composition rule for partial derivatives:



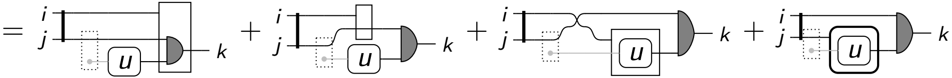


Because partial derivatives commute, the first term is zero.

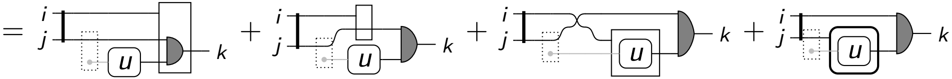
We expand the middle term using the composition rule for partial derivatives:



then use the product law on the middle term, and obtain

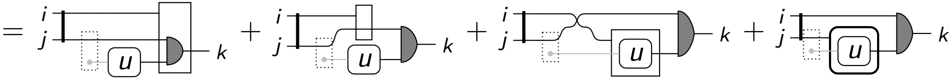


Proving the Ricci identity: cont.

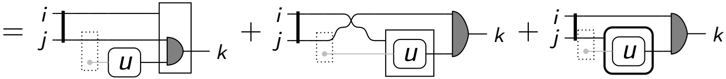


In the 2nd term, $\partial\,\mathrm{id} = 0$ so the whole term can be simplified away

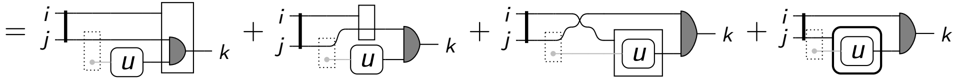
Proving the Ricci identity: cont.



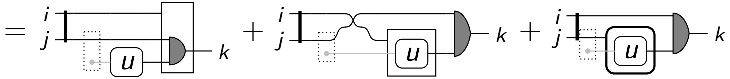
In the 2nd term, $\partial\,\mathrm{id} = 0$ so the whole term can be simplified away



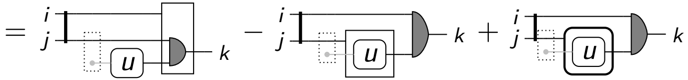
then we commute swap and antisymmetrisation, so the middle term changes sign:



In the 2nd term, $\partial\,\mathrm{id} = 0$ so the whole term can be simplified away

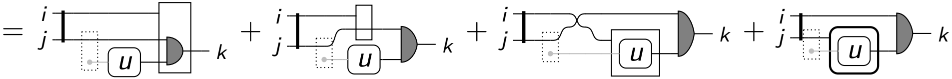


then we commute swap and antisymmetrisation, so the middle term changes sign:

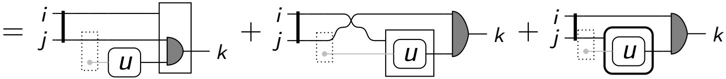


then we use the **Lemma** again to expand ∇u in the last term

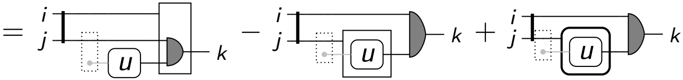
Proving the Ricci identity: cont.



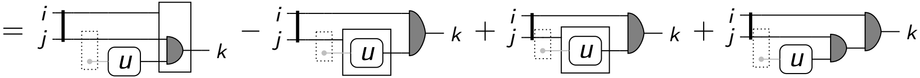
In the 2nd term, $\partial\,\mathrm{id} = 0$ so the whole term can be simplified away



then we commute swap and antisymmetrisation, so the middle term changes sign:

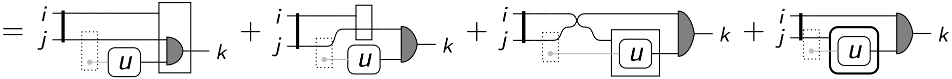


then we use the **Lemma** again to expand ∇u in the last term

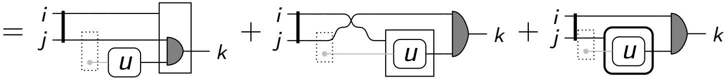


and note that the middle two terms cancel to yield:

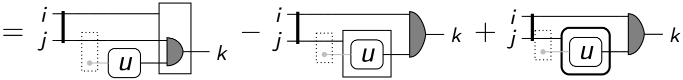
Proving the Ricci identity: cont.



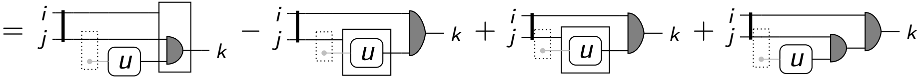
In the 2nd term, $\partial\,\mathrm{id} = 0$ so the whole term can be simplified away



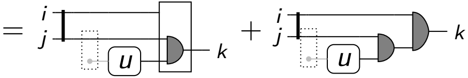
then we commute swap and antisymmetrisation, so the middle term changes sign:



then we use the **Lemma** again to expand ∇u in the last term



and note that the middle two terms cancel to yield:



which is the RHS (modulo some wiring).

[Optional] Riemann and Ricci tensors

The Ricci tensor is a contraction of the Riemann curvature used twice in the main equation:

$$\mathsf{ricci}, \mathsf{grLhs} :: (\mathsf{Additive}\ z, \mathsf{CoordinateCategory}\ z) \Rightarrow \mathsf{P}\ z\ \mathsf{r}\ T_z \multimap \mathsf{P}\ z\ \mathsf{r}\ T_z \multimap \mathsf{R}\ z\ \mathsf{r}$$
$$\mathsf{ricci}_{\ \mathsf{j}\ \mathsf{k}} = \mathsf{contract}\ (\lambda^{\ \mathsf{i}}_{\ \ \mathsf{i}} \to \mathsf{curvature}^{\ \mathsf{i}}_{\ \ \mathsf{j}\ \mathsf{i}\ \mathsf{k}})$$

Using it, the left-hand-side of the GenRelEq is

$$\mathsf{grLhs}_{\ \mathsf{i}\ \mathsf{j}} = \mathsf{plus}\ (\lambda\ \mathsf{c} \to \textbf{case}\ \mathsf{c}\ \textbf{of}$$
$$\quad \mathsf{False} \to \mathsf{ricci}_{\ \mathsf{i}\ \mathsf{j}}$$
$$\quad \mathsf{True}\ \to \mathsf{constant}\ (1\ /\ 2) \star \mathsf{contract}\ (\lambda^{\ \mathsf{k}}_{\ \ \mathsf{k}} \to \mathsf{ricci}_{\ \mathsf{k}}\ (\mathsf{lower}^{\ \mathsf{k}})) \star \mathsf{metric}_{\ \mathsf{i}\ \mathsf{j}})$$

We can then convert it to a morphism:

$$\mathsf{grLhsM} :: (\mathsf{Additive}\ z, \mathsf{CoordinateCategory}\ z) \Rightarrow (T_z \otimes T_z) \overset{z}{\leadsto} \mathbf{1}$$
$$\mathsf{grLhsM} = \mathsf{tensorEval}_1\ (\lambda\ \mathsf{k} \to \mathsf{split}\ \mathsf{k}\ \&\ \lambda\ (_{\mathsf{i}}, _{\mathsf{j}}) \to \mathsf{grLhs}_{\ \mathsf{i}\ \mathsf{j}})$$

Point-mass example in Albert: Step 1

The simplest example of a solution to the GenRelEq is the metric which describes the gravitational effects of a point-sized mass: the Schwarzschild metric.
We can express this metric and verify that it is a solution.

Point-mass example in ALBERT: Step 1

The simplest example of a solution to the GenRelEq is the metric which describes the gravitational effects of a point-sized mass: the Schwarzschild metric.

We can express this metric and verify that it is a solution.

Step 1: define the coordinate system.

Roughly speaking, Schwarzschild coordinates are spherical coordinates with an extra component for time.

$$\mathbf{data}\ \mathsf{Spherical} = \mathsf{Time} \mid \mathsf{Rho} \mid \mathsf{Theta} \mid \mathsf{Phi}$$

The point mass is located at the origin ($\rho{=}0$) at every point in time.

The T_{ij} tensor is zero everywhere except at the origin.

Step 2: Schwarzschild metric

The Schwarzschild metric is defined in terms of the considered mass M or alternatively by the Schwarzschild radius $r_s = 2GM/c^2$. Only the diagonal elements are non-zero.

$$
\begin{array}{l}
\mathsf{schwarzschild} :: \mathsf{Spherical} \to \mathsf{Spherical} \to \mathsf{S}_{\mathsf{Spherical}} \\
\mathsf{schwarzschild}\ \mathsf{Time}\ \ \mathsf{Time} \ = -(1 - r_s \mathbin{/} \mathsf{rho}) * (\mathsf{c} \mathbin{\hat{}} 2) \\
\mathsf{schwarzschild}\ \mathsf{Rho}\ \ \mathsf{Rho} \ = \ \ (1 - r_s \mathbin{/} \mathsf{rho}) \mathbin{\hat{}} (-1) \\
\mathsf{schwarzschild}\ \mathsf{Theta}\ \mathsf{Theta} = \mathsf{rho} \mathbin{\hat{}} 2 \\
\mathsf{schwarzschild}\ \mathsf{Phi}\ \ \mathsf{Phi} \ = (\mathsf{rho} * \sin \mathsf{theta}) \mathbin{\hat{}} 2 \\
\mathsf{schwarzschild}\ \_\ \ \_ \ = 0 \\
\mathsf{rho} \ \ = \mathsf{variable}\ \mathsf{Rho} \\
\mathsf{theta} = \mathsf{variable}\ \mathsf{Theta}
\end{array}
$$

We can then define the Matrix Spherical instance as follows:

$$
\begin{array}{l}
\textbf{instance}\ \mathsf{MetricCategory}\ M_{\mathsf{Spherical}}\ \textbf{where} \\
\quad \textbf{type}\ T_{M_{\mathsf{Spherical}}} = \mathsf{Atom}\ \mathsf{Spherical} \\
\quad g = \mathsf{Tab}\,(\lambda\,(\mathsf{Atom}\ \mathsf{i}, \mathsf{Atom}\ \mathsf{j})\ \_ \to \mathsf{schwarzschild}\ \mathsf{i}\ \mathsf{j})
\end{array}
$$

Step 3: Checking the main equation

We evaluate grLhsM with $z = \text{Matrix}\,\text{Spherical}$, and obtain a 4 by 4 matrix of symbolic expressions depending on Spherical coordinate variables.
We find that it simplifies to zero everywhere it is defined.
Thus we can verify that the Schwarzschild metric satisfies the general relativity equation.

Step 4: Interpretation

Looking back at the first two lines of the definition of the metric:

```
schwarzschild Time Time = - (1 - r_s / rho) * (c ^ 2)
schwarzschild Rho  Rho  =   (1 - r_s / rho) ^ (-1)
```

we note two singularities, one at the origin and one at $\rho = r_s$, the event horizon.

Step 4: Interpretation

Looking back at the first two lines of the definition of the metric:

schwarzschild Time Time $= -(1 - r_s$ / rho) $*$ (c ^ 2)

schwarzschild Rho Rho $=$ $(1 - r_s$ / rho) ^ (-1)

we note two singularities, one at the origin and one at $\rho = r_s$, the event horizon.

In classical (Newtonian) mechanics, the gravitational potential of a point mass is $\propto 1/rho$, which is also singular at the origin.

But here we also get a singularity at the Schwarzschild radius $r_s = 2GM/c^2$.

Step 4: Interpretation

Looking back at the first two lines of the definition of the metric:
schwarzschild Time Time $= -(1 - r_s / \text{rho}) * (\text{c} \,\hat{}\, 2)$
schwarzschild Rho Rho $= (1 - r_s / \text{rho}) \,\hat{}\, (-1)$
we note two singularities, one at the origin and one at $\rho = r_s$, the event horizon.
In classical (Newtonian) mechanics, the gravitational potential of a point mass is $\propto 1/rho$, which is also singular at the origin.
But here we also get a singularity at the Schwarzschild radius $r_s = 2GM/c^2$.
Every point mass is a black hole (some very small)!
(mass of our sun => 3km; of Earth => 9mm; moon => 0.1mm)

More about curved space-time

Imagine a small object (A) falling radially towards the black hole.

- The Schwarzschild coordinates are those of an observer (B) far away.
- In these coordinates, it takes infinite time for A to reach the event horison.

[1] after 6μs for a solar-mass black hole, after 10min for a supermassive black hole

More about curved space-time

Imagine a small object (A) falling radially towards the black hole.

- The Schwarzschild coordinates are those of an observer (B) far away.
- In these coordinates, it takes infinite time for A to reach the event horison.
- But for (A) the time to reach the horizon is finite!
- And it continuous through (at great speed) without problems.

[1] after 6μs for a solar-mass black hole, after 10min for a supermassive black hole

More about curved space-time

Imagine a small object (A) falling radially towards the black hole.

- The Schwarzschild coordinates are those of an observer (B) far away.
- In these coordinates, it takes infinite time for A to reach the event horison.
- But for (A) the time to reach the horizon is finite!
- And it continuous through (at great speed) without problems.
- The horizon is "just" a point of no return.

[1] after 6μs for a solar-mass black hole, after 10min for a supermassive black hole

More about curved space-time

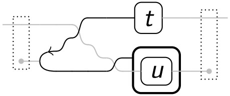
Imagine a small object (A) falling radially towards the black hole.

- The Schwarzschild coordinates are those of an observer (B) far away.
- In these coordinates, it takes infinite time for A to reach the event horison.
- But for (A) the time to reach the horizon is finite!
- And it continuous through (at great speed) without problems.
- The horizon is "just" a point of no return.
- And it will very soon[1] crash into the origin.

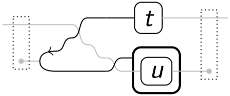
[1] after 6μs for a solar-mass black hole, after 10min for a supermassive black hole

Conclusions / contributions

- We provide two tensor EDSLs connected by automatic conversions: the index-based ALBERT and the morphism-based ROGER.
- The index-notation, diagram notation and matrices can be generated.

- We provide two tensor EDSLs connected by automatic conversions: the index-based ALBERT and the morphism-based ROGER.
- The index-notation, diagram notation and matrices can be generated.
- example $\mathrm{t\,u} = \mathrm{contract}\,(\lambda\,^{\mathrm{i}}\,_{\mathrm{i}} \rightarrow \mathrm{t}\,^{\mathrm{i}} \star \mathrm{deriv}\,_{\mathrm{i}}\,\mathrm{u})$ can either:
  1. render itself as the index-notation expression $t^i \nabla_i u$;
  2. render itself as the diagram 

  or
  3. run on matrix representations of the tensors t and u and compute the result

- We provide two tensor EDSLs connected by automatic conversions: the index-based ALBERT and the morphism-based ROGER.
- The index-notation, diagram notation and matrices can be generated.
- example $\mathrm{t}\ \mathrm{u} = \mathrm{contract}\ (\lambda\ {}^{\mathrm{i}}{}_{\mathrm{i}} \to \mathrm{t}^{\mathrm{i}} \star \mathrm{deriv}_{\mathrm{i}}\ \mathrm{u})$ can either:
  1. render itself as the index-notation expression $t^i \nabla_i u$;
  2. render itself as the diagram 

or
  3. run on matrix representations of the tensors t and u and compute the result
- Motivating example: Einstein's General Relativity equation for the curvature of space-time and the Schwarzschild metric tensor.