Tensorialを用いたテンソル解析の基礎

Mathematicaの数式演算の能力を人間能力では大変重労働になるテンソルの計算に活かしたいと思う人は少なくないはずである。
これをアシストするパッケージは古くからMathtensorがあり、本サイトでもすでに紹介している。
これより、安価で新しいMathematicaに対応したものにTensorialがあり2014年初でそのバージョンはVer5である。
他にも無償でOpenSourceのxActが開発されていてこれはMathTensorに近く非常に多くの関数やツールを持っている。
本サイトにxActの紹介も別にしてある。多様体を定義することからはじまり、スピンなども扱える。

一方でTensorialは軽快にMathematiaの力を利用できる特徴がある。Tensorialの古いバージョンはウルフラムサイトに無償でアーカイブされているので試してみるとよい。
最新のTensorial情報や購入は下記のサイトを見るとよい。
また、Mathematicaの基本操作に慣れていない人は本サイトのMathematicaの基礎等を一読しておくといいだろう。
本稿はまだ未完成で今後変更、加筆される。

http : // www.jfgouyet.fr / tcm / tcm.html

上記サイトによると残念ながら最新版は有償のようである。Tensorial4は100＄、TContinuumMechanics2_6は50$いる。
前者だけでもテンソルの演算はできる。

ガイドブックへのリンク **Tensorial Guide Page Link**

Tensorial5のインストールとロード

インストールするにはサイトからダウンロードしたファイルをProgramFilesにあるMathematicaのフォルダの中のAddOnsフォルダの中のPackagesにTensorCalculus5というフォルダを解凍し、ダウンロードファイルをそのまま入れる。ただ、Windows8以降ではこのフォルダには書き換え禁止の属性がつくのでいろいろいじりたい人は属性を解除するか隠しフォルダになっているDocuments_and_Settingの中のAppDataのMathematicaを探していれるとよい。
ロードするにはMathematicaを起動し、ノートブックを開き、MathematicaVer9では次のようにコマンドをいれる。
このコマンドはどの章においてもはじめにおこなう必要がある。
それではまず順に実行しながら基本的なコマンドを学ぼう。

```
Unprotect[SyntaxInformation];
Unprotect[Symmetric];
Unprotect[TensorSymmetry];
Needs["TensorCalculus5`Tensorial`"]
(*Needs["TensorCalculus4V6`Tensorial`"] *)
(*Needs["TContinuumMechanics21`TContinuumMechanics`"] *)
```

Tensorial5の基本
特殊相対性理論入門

ここからはTensrialはロードした直後で、何も定義していないとする。
はじめに次のように4次元ミンコフスキー時空を定義し、テンソルを定義する。

```
DeclareBaseIndices[{0, 1, 2, 3}]
DefineTensor[x, x', a, b, v, η]
```

ローレンツ変換

次にミンコフスキー時空の計量を次で定義する。

```
metric = DiagonalMatrix[{-1, 1, 1, 1}];
% // MatrixForm
```

$$\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Tensorialでは定義したルールを次のように記号に代入し、テンソルを決める

```
SetTensorValues[η@dd[i, j], metric]
```

ToArrayValues[]を使って中身を確認すると

```
η@dd[i, j]
% // ToArrayValues[] // MatrixForm
```

$$\eta_{i\,j}$$

$$\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

x'系がxの一次変換で結ばれるとしてEinsteinSum[]を利用して次のように同じ添え字については和をとることができる。

EinsteinSum[]

```
x'@u[μ] = a@ud[μ, ν] x@u[ν] + b@u[μ] // EinsteinSum[];
% // MatrixForm
x'@u[ν] = a@ud[ν, μ] x@u[μ] + b@u[ν] // EinsteinSum[];
% // MatrixForm
```

$$b^{\mu} + a^{\mu}{}_{0}\, x^{0} + a^{\mu}{}_{1}\, x^{1} + a^{\mu}{}_{2}\, x^{2} + a^{\mu}{}_{3}\, x^{3}$$

$$b^{\nu} + a^{\nu}{}_{0}\, x^{0} + a^{\nu}{}_{1}\, x^{1} + a^{\nu}{}_{2}\, x^{2} + a^{\nu}{}_{3}\, x^{3}$$

この1次変換が特殊相対論の要請を満たせば $a^{\mu}{}_{\nu}$はローレンツ変換であり、
次の2つのスカラーは等しい。

```
η@dd[μ, ν] x@u[μ] x@u[ν]
% // ToArrayValues[]
η@dd[μ, ν] x'@u[μ] x'@u[ν]
```

$$x^{\mu}\, x^{\nu}\, \eta_{\mu\,\nu}$$

$$-\left(x^{0}\right)^{2} + \left(x^{1}\right)^{2} + \left(x^{2}\right)^{2} + \left(x^{3}\right)^{2}$$

$$\left(b^{\mu} + a^{\mu}{}_{0}\, x^{0} + a^{\mu}{}_{1}\, x^{1} + a^{\mu}{}_{2}\, x^{2} + a^{\mu}{}_{3}\, x^{3}\right)\left(b^{\nu} + a^{\nu}{}_{0}\, x^{0} + a^{\nu}{}_{1}\, x^{1} + a^{\nu}{}_{2}\, x^{2} + a^{\nu}{}_{3}\, x^{3}\right)\eta_{\mu\,\nu}$$

係数を比較するとローレンツ変換になるためには次の関係が成り立つ。

$$\eta_{\mu\nu} = \eta_{\rho\sigma}\, a^{\rho}{}_{\mu}\, a^{\sigma}{}_{\nu} \tag{1}$$

これは次のようにも表現できる。

$$\eta_{\mu\nu} = \eta_{\rho\sigma}\, a^{\rho}{}_{\mu}\, a^{\sigma}{}_{\nu} \tag{2}$$

右辺は次のように計算されるので左辺の行列要素と比較すると次のようになる。

```
η@dd[ρ, σ] a@ud[ρ, μ] a@ud[σ, ν]
fg1 = % // ToArrayValues[] // Simplify // MatrixForm
```

$$a^{\rho}{}_{\mu}\, a^{\sigma}{}_{\nu}\, \eta_{\rho\sigma}$$

$$\begin{pmatrix} -\left(a^0{}_0\right)^2+\left(a^1{}_0\right)^2+\left(a^2{}_0\right)^2+\left(a^3{}_0\right)^2 & -a^0{}_0\,a^0{}_1+a^1{}_0\,a^1{}_1+a^2{}_0\,a^2{}_1+a^3{}_0\,a^3{}_1 & -a^0{}_0\,a^0{}_2+a^1{}_0\,a^1{}_2+a^2 \\ -a^0{}_0\,a^0{}_1+a^1{}_0\,a^1{}_1+a^2{}_0\,a^2{}_1+a^3{}_0\,a^3{}_1 & -\left(a^0{}_1\right)^2+\left(a^1{}_1\right)^2+\left(a^2{}_1\right)^2+\left(a^3{}_1\right)^2 & -a^0{}_1\,a^0{}_2+a^1{}_1\,a^1{}_2+a^2 \\ -a^0{}_0\,a^0{}_2+a^1{}_0\,a^1{}_2+a^2{}_0\,a^2{}_2+a^3{}_0\,a^3{}_2 & -a^0{}_1\,a^0{}_2+a^1{}_1\,a^1{}_2+a^2{}_1\,a^2{}_2+a^3{}_1\,a^3{}_2 & -\left(a^0{}_2\right)^2+\left(a^1{}_2\right)^2+\left(a \\ -a^0{}_0\,a^0{}_3+a^1{}_0\,a^1{}_3+a^2{}_0\,a^2{}_3+a^3{}_0\,a^3{}_3 & -a^0{}_1\,a^0{}_3+a^1{}_1\,a^1{}_3+a^2{}_1\,a^2{}_3+a^3{}_1\,a^3{}_3 & -a^0{}_2\,a^0{}_3+a^1{}_2\,a^1{}_3+a^2 \end{pmatrix}$$

対角要素だけをみれば、次の関係式が得られる。

$$a^0{}_0 = \pm\sqrt{1+\sum_{k=1}^{3}\left(a^k{}_0\right)^2} \tag{3}$$

これから

$$a^0{}_0 \geq 1 \text{ or } a^0{}_0 \leq -1 \tag{4}$$

さらに行列式|a|=±1 を満たす必要があるから、この４通りのうち次を満たすものを固有ローレンツ変換(poper Lorentz transformation) という。

$$a^0{}_0 \geq 1 \quad |a| = 1 \tag{5}$$

*Mathematica*を利用すると計算の面倒さはある程度考えなくてもよいのが長所ではあるが、経験的にやはり手計算で見通しをもっておこないと大きなミスをすることがあるのでTnsorialもMathemtiaもあくまで道具であることを知る必要がある。そのためにはその仕組みと振る舞いをよく知っておく必要がある。

実はTensorial は次のように上記の条件を満たすローレンツ変換の関数を持っている。

速さの比β=V/c、速度の向きを{1,0,0}として

```
LorentzTransformation[β, {1, 0, 0}] // MatrixForm
```

さらに光速の7割の速さとして方向を{1,1,1}に変えると

```
LorentzTransformation[.7, {1, 1, 1}] // MatrixForm
```

もっとも一般的に{n1,n2,n3}の向きに速度比βとして

以下の例では2行目で前の結果を速度比βを1未満、向きを表すnは実数であるとして簡単化させ、LT変数に代入。

最後の行で行列表示させてある。

Mathematica は$\sqrt{}$をとらせると条件表示が出たり、行列表示をさせると正しく代入できないので注意する。

```
LorentzTransformation[β, {n1, n2, n3}] ;
LT = Simplify[%, 0 < β < 1 ⋀ {n1, n2, n3} ∈ Reals];
% // MatrixForm
```

$$\begin{pmatrix} \frac{1}{\sqrt{1-\beta^2}} & -\frac{n1\,\beta}{\sqrt{-\left(n1^2+n2^2+n3^2\right)\left(-1+\beta^2\right)}} & -\frac{n2\,\beta}{\sqrt{-\left(n1^2+n2^2+n3^2\right)\left(-1+\beta^2\right)}} & -\frac{n3\,\beta}{\sqrt{-\left(n1^2+n2^2+n3^2\right)\left(-1+\beta^2\right)}} \\ -\frac{n1\,\beta}{\sqrt{-\left(n1^2+n2^2+n3^2\right)\left(-1+\beta^2\right)}} & 1+\frac{n1^2\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{n1^2+n2^2+n3^2} & \frac{n1\,n2\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{n1^2+n2^2+n3^2} & \frac{n1\,n3\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{n1^2+n2^2+n3^2} \\ -\frac{n2\,\beta}{\sqrt{-\left(n1^2+n2^2+n3^2\right)\left(-1+\beta^2\right)}} & \frac{n1\,n2\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{n1^2+n2^2+n3^2} & 1+\frac{n2^2\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{n1^2+n2^2+n3^2} & \frac{n2\,n3\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{n1^2+n2^2+n3^2} \\ -\frac{n3\,\beta}{\sqrt{-\left(n1^2+n2^2+n3^2\right)\left(-1+\beta^2\right)}} & \frac{n1\,n3\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{n1^2+n2^2+n3^2} & \frac{n2\,n3\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{n1^2+n2^2+n3^2} & 1+\frac{n3^2\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{n1^2+n2^2+n3^2} \end{pmatrix}$$

これがローレンツ条件式2を満たすか確認しよう。

留意すべきは他のTensorソフトもそうだが表現η@dd[i,j]は行列形式になっているわけではないとうことである。

しかし、Tensorialでは下記のようにToArrayValues[]を使って行列にできるのでこれを適当な変数(YT)に代入する。

```
YT = η@dd[i, j] // ToArrayValues[];
Transpose[LT].YT.LT // Simplify // MatrixForm
```

$$\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

おみごと！対角化され条件を満たした。

せっかくローレンツ変換の一般式が得られたのでそこでこれをMathematica　のManipulateを使って動的に視覚化してみよう。

独立して使えるようにLTに代入した一般ローレンツ変換を以下の例では関数として定義している。

静止系では簡単な球がローレンツ変換により光速に近い速さになるとどんなふうに見えるか確かめてみる。

```
Manipulate[
 r0 = {{1, x, y, z}};
 kc = 1.4;
 g1 = Graphics3D[{Arrow[{{-2, 0, 0}, {2, 0, 0}}], Arrow[{{0, -2, 0}, {0, 2, 0}}],
    Arrow[{{0, 0, -2}, {0, 0, 2}}],
    Arrow[Tube[{kc {nx v, ny v, nz v}, 2 kc {nx v, ny v, nz v}}]]};
 g2 = ContourPlot3D[x^2 + y^2 + z^2 == 1, {x, -1, 1}, {y, -1, 1}, {z, -1, 1},
   ContourStyle → Opacity[0.3], Mesh → None];
 g3 =
  ContourPlot3D[
   (Part[r0.Lz[v, nx, ny, nz], 1][[2]]) ^2 + (Part[r0.Lz[v, nx, ny, nz], 1][[3]]) ^2 +
     (Part[r0.Lz[v, nx, ny, nz], 1][[4]]) ^2 == 1, {x, -2, 2}, {y, -2, 2},
   {z, -2, 2}, Mesh → None];
 Show[g1, g2, g3],
 {{v, 0, "速さの比"}, 0, 0.99, 0.01, Appearance → "Labeled"},
 {{nx, 0.5, "x方向"}, -1, 1, 0.1, Appearance → "Labeled"},
 {{ny, 0.5, "y方向"}, -1, 1, 0.1, Appearance → "Labeled"},
 {{nz, 0.5, "z方向"}, -1, 1, 0.1, Appearance → "Labeled"},
 Initialization :>
  (
   {Lz[β_, n1_, n2_, n3_] :=
```

$$\left\{\left\{\frac{1}{\sqrt{1-\beta^2}}, -\frac{n1\,\beta}{\sqrt{1-\beta^2}\,\sqrt{Abs[n1]^2+Abs[n2]^2+Abs[n3]^2}},\right.\right.$$

$$\left.-\frac{n2\,\beta}{\sqrt{1-\beta^2}\,\sqrt{Abs[n1]^2+Abs[n2]^2+Abs[n3]^2}},\right.$$

$$\left.\left.-\frac{n3\,\beta}{\sqrt{1-\beta^2}\,\sqrt{Abs[n1]^2+Abs[n2]^2+Abs[n3]^2}}\right\},\right.$$

$$\left\{-\frac{n1\,\beta}{\sqrt{1-\beta^2}\,\sqrt{Abs[n1]^2+Abs[n2]^2+Abs[n3]^2}}, 1+\frac{n1^2\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{Abs[n1]^2+Abs[n2]^2+Abs[n3]^2},\right.$$

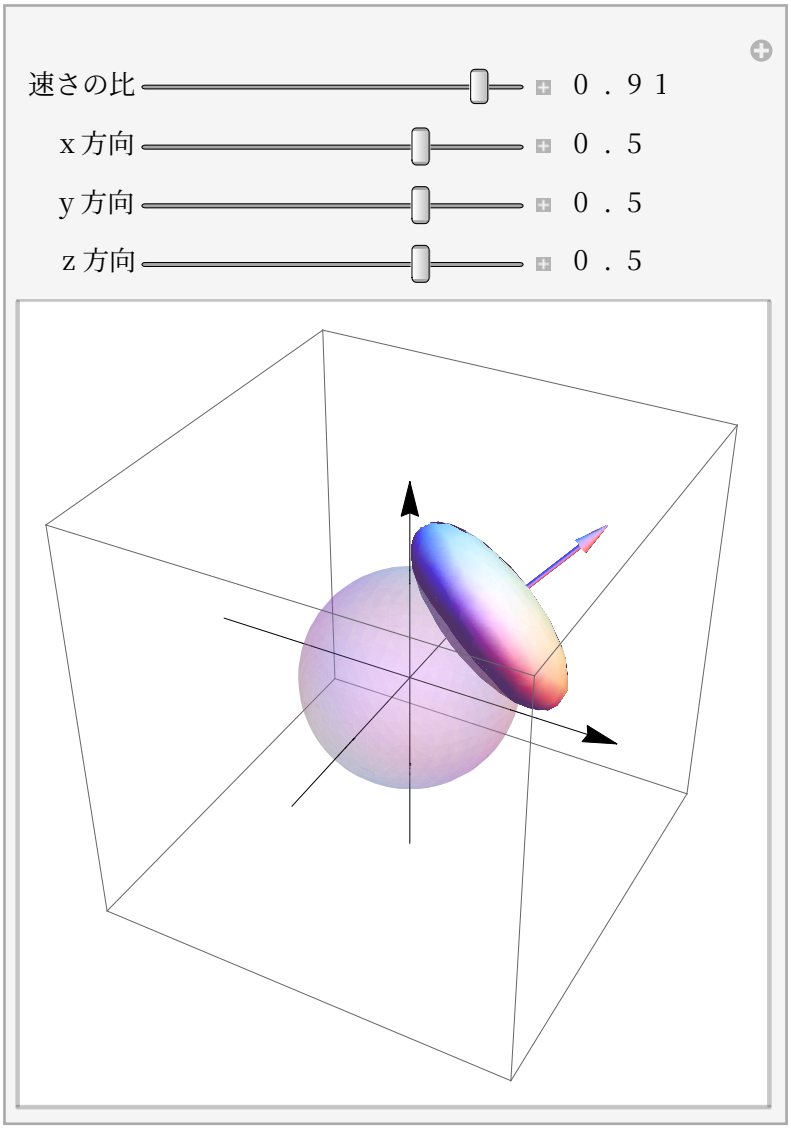
$$\frac{\text{n1 n2}\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{\text{Abs[n1]}^2+\text{Abs[n2]}^2+\text{Abs[n3]}^2},\ \frac{\text{n1 n3}\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{\text{Abs[n1]}^2+\text{Abs[n2]}^2+\text{Abs[n3]}^2}\Big\},$$

$$\Big\{-\frac{\text{n2}\ \beta}{\sqrt{1-\beta^2}\ \sqrt{\text{Abs[n1]}^2+\text{Abs[n2]}^2+\text{Abs[n3]}^2}},\ \frac{\text{n1 n2}\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{\text{Abs[n1]}^2+\text{Abs[n2]}^2+\text{Abs[n3]}^2},$$

$$1+\frac{\text{n2}^2\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{\text{Abs[n1]}^2+\text{Abs[n2]}^2+\text{Abs[n3]}^2},\ \frac{\text{n2 n3}\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{\text{Abs[n1]}^2+\text{Abs[n2]}^2+\text{Abs[n3]}^2}\Big\},$$

$$\Big\{-\frac{\text{n3}\ \beta}{\sqrt{1-\beta^2}\ \sqrt{\text{Abs[n1]}^2+\text{Abs[n2]}^2+\text{Abs[n3]}^2}},\ \frac{\text{n1 n3}\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{\text{Abs[n1]}^2+\text{Abs[n2]}^2+\text{Abs[n3]}^2},$$

$$\frac{\text{n2 n3}\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{\text{Abs[n1]}^2+\text{Abs[n2]}^2+\text{Abs[n3]}^2},\ 1+\frac{\text{n3}^2\left(-1+\frac{1}{\sqrt{1-\beta^2}}\right)}{\text{Abs[n1]}^2+\text{Abs[n2]}^2+\text{Abs[n3]}^2}\Big\}\Big\}\Big\}\Big\}\Big]$$



このように簡単にTensor計算の結果をMathematicaで利用できるのはTensorialのメリットである。

電磁気学入門

ここからもTensrialはロードした直後で、何も定義していないとする。

はじめに次のように4次元ミンコフスキー時空を定義する。次に4次元の基底、テンソルを次のように定義しよう。

最後のコマンドでクロネッカーδを定義している。

```
DeclareBaseIndices[{1, -1, -1, -1}]
DeclareBaseIndices[{0, 1, 2, 3}]
labs = {x, δ, g, Γ};
DefineTensor[a, b, x, x', J, ξ, g, F, Ef, Bf, A, δ, η, Λ, F, Γ]
SetTensorValues[δ@ud[i, j], IdentityMatrix[NDim]]
```

まず、計量は平坦にとる。Tensorialのメリットは行列要素をそのまま代入できることである。

行列演算もそのままできるので取扱いが楽である。xActやMathTensorにはこうした軽快さはなく、表現と中身を常に別に考えておかないといけない。

```
(cmetric = DiagonalMatrix[{-1, 1, 1, 1}]) // MatrixForm;
(metric = cmetric // CoordinatesToTensors[{t, x, y, z}]) // MatrixForm
```

$$\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

MathematicaのMapThreadを用いて上の結果から計量テンソルgを定義する。

```
MapThread[SetTensorValues[#1, #2] &,
  {{g@dd[a, b], g@uu[a, b]}, {metric, Inverse[metric] // Simplify}}];
```

ToArrayValues[]を利用すると和をつくり次のように表現される。

```
g@uu[i, j]
% // ToArrayValues[] // MatrixForm
g@dd[i, j]
fg = % // ToArrayValues[] // MatrixForm
g@uu[i, j] g@dd[i, j] // ToArrayValues[]
```

$$g^{i\,j}$$

$$\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$g_{i\,j}$$

$$\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
4
```

これでユークリッド空間の計量テンソルが次のように定義できた。

*MetricSimplify[*計量テンソル*]*

縮約をつかうにはMetricSimplify[]を使う

```
g@dd[i, k] g@uu[k, j]
% // MetricSimplify[g]
```

$$g^{k\,j}\, g_{i\,k}$$

$$g_{i}^{\ j}$$

電磁場のテンソル

Efを電場の4元ベクトル、Bfを磁場の4元ベクトル、Fを電磁場の4元ベクトルとする。
0はt,1はx,2はy,3はzに対応し、次のように偏微分をもちいてまずEFを定義する。
EinsteinArray[{1,2,3}]を使うと成分を全て表現できるのは便利である。

```
Ef@d[α] == PartialD[labs][A@d[0], x@u[α]] - PartialD[labs][A@d[α], x@u[0]]
% // EinsteinArray[{1, 2, 3}] // MatrixForm
```

$$\mathrm{Ef}_{\alpha} == \frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^{\alpha}} - \frac{\partial \mathrm{A}_{\alpha}}{\partial \mathrm{x}^0}$$

$$\begin{pmatrix} \mathrm{Ef}_1 == \frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^1} - \frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^0} \\ \mathrm{Ef}_2 == \frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^2} - \frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^0} \\ \mathrm{Ef}_3 == \frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^3} - \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^0} \end{pmatrix}$$

同様に次のコマンドで磁場テンソルを定義する。

```
Bf@dd[β, γ] == PartialD[labs][A@d[γ], x@u[β]] - PartialD[labs][A@d[β], x@u[γ]]
% // EinsteinArray[{1, 2, 3}] // MatrixForm
```

$$\mathrm{Bf}_{\beta\gamma} == -\frac{\partial \mathrm{A}_{\beta}}{\partial \mathrm{x}^{\gamma}} + \frac{\partial \mathrm{A}_{\gamma}}{\partial \mathrm{x}^{\beta}}$$

$$\begin{pmatrix} \mathrm{Bf}_{10} == \frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^1} - \frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^0} & \mathrm{Bf}_{11} == 0 & \mathrm{Bf}_{12} == -\frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^2} + \frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^1} & \mathrm{Bf}_{13} == -\frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^3} + \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^1} \\ \mathrm{Bf}_{20} == \frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^2} - \frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^0} & \mathrm{Bf}_{21} == \frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^2} - \frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^1} & \mathrm{Bf}_{22} == 0 & \mathrm{Bf}_{23} == -\frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^3} + \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^2} \\ \mathrm{Bf}_{30} == \frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^3} - \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^0} & \mathrm{Bf}_{31} == \frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^3} - \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^1} & \mathrm{Bf}_{32} == \frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^3} - \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^2} & \mathrm{Bf}_{33} == 0 \end{pmatrix}$$

ベクトルポテンシャルを用いた電磁場のテンソルは全体で次のように定義できる。

```
F@dd[β, γ] == PartialD[labs][A@d[γ], x@u[β]] - PartialD[labs][A@d[β], x@u[γ]]
% // ToArrayValues[] // MatrixForm
```

$$\mathrm{F}_{\beta\gamma} == -\frac{\partial \mathrm{A}_{\beta}}{\partial \mathrm{x}^{\gamma}} + \frac{\partial \mathrm{A}_{\gamma}}{\partial \mathrm{x}^{\beta}}$$

$$\begin{pmatrix} \mathrm{F}_{00} == 0 & \mathrm{F}_{01} == -\frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^1} + \frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^0} & \mathrm{F}_{02} == -\frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^2} + \frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^0} & \mathrm{F}_{03} == -\frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^3} + \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^0} \\ \mathrm{F}_{10} == \frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^1} - \frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^0} & \mathrm{F}_{11} == 0 & \mathrm{F}_{12} == -\frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^2} + \frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^1} & \mathrm{F}_{13} == -\frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^3} + \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^1} \\ \mathrm{F}_{20} == \frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^2} - \frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^0} & \mathrm{F}_{21} == \frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^2} - \frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^1} & \mathrm{F}_{22} == 0 & \mathrm{F}_{23} == -\frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^3} + \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^2} \\ \mathrm{F}_{30} == \frac{\partial \mathrm{A}_0}{\partial \mathrm{x}^3} - \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^0} & \mathrm{F}_{31} == \frac{\partial \mathrm{A}_1}{\partial \mathrm{x}^3} - \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^1} & \mathrm{F}_{32} == \frac{\partial \mathrm{A}_2}{\partial \mathrm{x}^3} - \frac{\partial \mathrm{A}_3}{\partial \mathrm{x}^2} & \mathrm{F}_{33} == 0 \end{pmatrix}$$

Tensorialは行列からテンソルを簡単につくることができる特徴があるので
相対論的な電磁気でよく利用されるローレンツ変換や、電磁場テンソルを行列で定義する。

$$\text{trans} = \begin{pmatrix} \gamma & -\gamma\beta & 0 & 0 \\ -\gamma\beta & \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix};$$

$$\text{itrans} = \begin{pmatrix} \gamma & \gamma\beta & 0 & 0 \\ \gamma\beta & \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix};$$

$$\text{fbasis} = \begin{pmatrix} 0 & -E1 & -E2 & -E3 \\ E1 & 0 & -B3 & B2 \\ E2 & B3 & 0 & -B1 \\ E3 & -B2 & B1 & 0 \end{pmatrix};$$

$$\text{finverse} = \begin{pmatrix} 0 & E1 & E2 & E3 \\ -E1 & 0 & -B3 & B2 \\ -E2 & B3 & 0 & -B1 \\ -E3 & -B2 & B1 & 0 \end{pmatrix};$$

$$\text{fdual} = \begin{pmatrix} 0 & -B1 & -B2 & -B3 \\ B1 & 0 & E3 & -E2 \\ B2 & -E3 & 0 & E1 \\ B3 & E2 & -E1 & 0 \end{pmatrix};$$

*SetTensorValueRules[*計量テンソル、代入配列*]*

*TensorValueRules[*テンソル*]*

この定義をテンソルに次のコマンドで適用させる。このハンドリングのよさはTensorialの特徴である。

SetTensorValueRules[]で行列からテンソルを定義し、その置き換えルールをTensorValueRules[]で確認する。

```
SetTensorValueRules[Λ@ud[α, β], trans]
SetTensorValueRules[F@uu[α, β], fbasis]
SetTensorValueRules[F@dd[γ, η], finverse]
SetTensorValueRules[F@uu[α, β], fdual]
```

適用させたコマンドが次のように確認できる。

```
TensorValueRules[Λ]
TensorValueRules[F]
TensorValueRules[F]
```

$$\{\Lambda^0{}_0 \to \gamma,\ \Lambda^0{}_1 \to -\beta\gamma,\ \Lambda^0{}_2 \to 0,\ \Lambda^0{}_3 \to 0,\ \Lambda^1{}_0 \to -\beta\gamma,\ \Lambda^1{}_1 \to \gamma,\ \Lambda^1{}_2 \to 0,$$
$$\Lambda^1{}_3 \to 0,\ \Lambda^2{}_0 \to 0,\ \Lambda^2{}_1 \to 0,\ \Lambda^2{}_2 \to 1,\ \Lambda^2{}_3 \to 0,\ \Lambda^3{}_0 \to 0,\ \Lambda^3{}_1 \to 0,\ \Lambda^3{}_2 \to 0,\ \Lambda^3{}_3 \to 1\}$$

$$\{F^{00} \to 0,\ F^{01} \to -E1,\ F^{02} \to -E2,\ F^{03} \to -E3,\ F^{10} \to E1,\ F^{11} \to 0,\ F^{12} \to -B3,\ F^{13} \to B2,$$
$$F^{20} \to E2,\ F^{21} \to B3,\ F^{22} \to 0,\ F^{23} \to -B1,\ F^{30} \to E3,\ F^{31} \to -B2,\ F^{32} \to B1,\ F^{33} \to 0,$$
$$F_{00} \to 0,\ F_{01} \to E1,\ F_{02} \to E2,\ F_{03} \to E3,\ F_{10} \to -E1,\ F_{11} \to 0,\ F_{12} \to -B3,\ F_{13} \to B2,$$
$$F_{20} \to -E2,\ F_{21} \to B3,\ F_{22} \to 0,\ F_{23} \to -B1,\ F_{30} \to -E3,\ F_{31} \to -B2,\ F_{32} \to B1,\ F_{33} \to 0\}$$

$$\{\mathcal{F}^{00} \to 0,\ \mathcal{F}^{01} \to -B1,\ \mathcal{F}^{02} \to -B2,\ \mathcal{F}^{03} \to -B3,\ \mathcal{F}^{10} \to B1,\ \mathcal{F}^{11} \to 0,\ \mathcal{F}^{12} \to E3,\ \mathcal{F}^{13} \to -E2,$$
$$\mathcal{F}^{20} \to B2,\ \mathcal{F}^{21} \to -E3,\ \mathcal{F}^{22} \to 0,\ \mathcal{F}^{23} \to E1,\ \mathcal{F}^{30} \to B3,\ \mathcal{F}^{31} \to E2,\ \mathcal{F}^{32} \to -E1,\ \mathcal{F}^{33} \to 0\}$$

ローレンツブーストは次のようになる。

```
Λ@ud[i, j]
% // ToArrayValues[] // MatrixForm
```

$$\Lambda^{i}{}_{j}$$

$$\begin{pmatrix} \gamma & -\beta\gamma & 0 & 0 \\ -\beta\gamma & \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

座標にに作用させ、成分を得たい場合は

```
Λ@ud[i, j] x@u[j]
% // ToArrayValues[] // MatrixForm
```

$$x^{j}\,\Lambda^{i}{}_{j}$$

$$\begin{pmatrix} \gamma\,x^0 - \beta\gamma\,x^1 \\ -\beta\gamma\,x^0 + \gamma\,x^1 \\ x^2 \\ x^3 \end{pmatrix}$$

電磁場テンソルはは次のようになる。

```
F@uu[i, j]
% // ToArrayValues[] // MatrixForm
```

$$F^{i\,j}$$

$$\begin{pmatrix} 0 & -E1 & -E2 & -E3 \\ E1 & 0 & -B3 & B2 \\ E2 & B3 & 0 & -B1 \\ E3 & -B2 & B1 & 0 \end{pmatrix}$$

この F^{ij}から g_{ij} を作用させ、下付の F_{ij}を作り、行列表示させる。
次のように定義したものと同じ結果になることがわかる。

```
g@dd[α, γ] g@dd[η, β] F@uu[γ, η]
% // MetricSimplify[g]
%% // ToArrayValues[] // MatrixForm
F@dd[α, β] // ToArrayValues[] // MatrixForm
```

$$F^{\gamma\,\eta}\,g_{\alpha\,\gamma}\,g_{\eta\,\beta}$$

$$F_{\alpha\,\beta}$$

$$\begin{pmatrix} 0 & E1 & E2 & E3 \\ -E1 & 0 & -B3 & B2 \\ -E2 & B3 & 0 & -B1 \\ -E3 & -B2 & B1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & E1 & E2 & E3 \\ -E1 & 0 & -B3 & B2 \\ -E2 & B3 & 0 & -B1 \\ -E3 & -B2 & B1 & 0 \end{pmatrix}$$

この F^{ij}から内積をつくりスカラー値を得てみよう。

```
g@dd[α, γ] g@dd[η, β] F@uu[γ, η] F@uu[α, β]
% // MetricSimplify[g]
%% // ToArrayValues[]
T1 = Simplify[%]
```

$$F^{\alpha\beta}\, F^{\gamma\eta}\, g_{\alpha\gamma}\, g_{\eta\beta}$$

$$F_{\gamma\eta}\, F^{\gamma\eta}$$

$$2\,B1^2 + 2\,B2^2 + 2\,B3^2 - 2\,E1^2 - 2\,E2^2 - 2\,E3^2$$

$$2\left(B1^2 + B2^2 + B3^2 - E1^2 - E2^2 - E3^2\right)$$

さらに時間成分の和を4分の1して引けばエネルギーが得られる。

```
g@dd[0, 0] g@dd[η, β] F@uu[0, η] F@uu[0, β]
% // MetricSimplify[g]
%% // ToArrayValues[]
T2 = Simplify[%]
Simplify[T1 / 4 - T2]
```

$$F^{0\beta}\, F^{0\eta}\, g_{0\,0}\, g_{\eta\beta}$$

$$F^{0}{}_{\eta}\, F^{0\eta}\, g_{0\,0}$$

$$-E1^2 - E2^2 - E3^2$$

$$-E1^2 - E2^2 - E3^2$$

$$\frac{1}{2}\left(B1^2 + B2^2 + B3^2 + E1^2 + E2^2 + E3^2\right)$$