InteriorProduct (⊖)

`InteriorProduct[x, y]`
denotes the interior (or CircleMinus) product of the expressions `x` and `y`.

Details

- The `InteriorProduct` is `Listable`.
- The interior product is a binary operation, and is not associative (except for a few special cases) like the exterior, regressive and Clifford products.
- The `CircleMinus` symbol ⊖ may be entered by using the *Basic Operations* palette or the sequence ESCc-ESC.
- An interior product □⊖□ may be entered by using the *Basic Operations* palette. Continued *nested* interior products may be entered by continued clicking on the □⊖□ button. For example, three successive clicks produces □⊖□⊖□⊖□. *Mathematica* parses this as ((□⊖□)⊖□)⊖□. This parsing conforms naturally to properties of the interior product, and is indeed part of the reason why the `CircleMinus` operator was chosen to denote the interior product. The □ are placeholders in which expressions may be entered successively using the Tab key.
- An interior product of two expressions of the same (single) grade is termed an *inner product*. An inner product of two expressions of grade 1 is termed a *scalar product*.
- The interior product may be considered a type of tensor contraction, reducing the grade of the left hand element by the grade of the right hand element. If a lower grade is on the left the result is zero.
- The interior products can be simplified with `SimplifyInteriorProducts`, `ExpandInteriorProducts`, `ExpandAndSimplifyInteriorProducts` or ★$\mathcal{G}$.
- When `Basis` elements are present in the expressions interior products can be further evaluated with `ToMetricElements`, which also performs the above simplifications.
- The interior product implies the use of a metric.
- The interior product is defined in terms of the regressive product and GrassmannComplement: $x \ominus y == x \vee \overline{y}$, as discussed in Chapter 6 of the book.
- Interior products can usually be evaluated more quickly using the `ContractInteriorProducts` routine. Using the metric, this converts the interior product to a Contractor of a vector on a form, which evaluates much faster.

Examples (1)

Basic Examples (1)

In[1]:=
```
<< GrassmannCalculus`
```

In[2]:=
```
★A
```

These inputs are equivalent:

In[3]:=
```
{InteriorProduct[x, y], CircleMinus[x, y], x⊖y}
```
Out[3]= {x⊖y, x⊖y, x⊖y}

The interior product is listable.

In[4]:=
```
u⊖{w, z}
{u, v}⊖{w, z}
{{u, v}, {x, y}}⊖z
{{u, v}, {x, y}}⊖{w, z}
```
Out[4]= {u⊖w, u⊖z}

Out[4]= {u⊖w, v⊖z}

Out[4]= {{u⊖z, v⊖z}, {x⊖z, y⊖z}}

Out[4]= {{u⊖w, v⊖w}, {x⊖z, y⊖z}}

Note that the last is not a matrix product.

Some specific examples: The following two examples contract a trivector with a bivector to produce a vector.

In[5]:=
```
★P
e1∧e2∧e3⊖e1∧e2
% // ToMetricElements
```
Out[5]= $(e_1 \wedge e_2 \wedge e_3) \ominus (e_1 \wedge e_2)$

Out[5]= e_3

In[6]:=
```
(e1 + 2 e2 + 3 e3) ∧ (e2 - e3) ∧ (a e1 + b e2 + e3) ⊖ e1 ∧ e2
% // ToMetricElements
```
Out[6]= $((e_1 + 2 e_2 + 3 e_3) \wedge (e_2 - e_3) \wedge (a e_1 + b e_2 + e_3)) \ominus (e_1 \wedge e_2)$

Out[6]= $(1 - 5 a + b) e_3$

The following is an example of a scalar product of two vectors.

In[7]:=
```
{1, 1, 1}.Basis⊖{1, 2, 3}.Basis
% // ToMetricElements
```
Out[7]= $(e_1 + e_2 + e_3) \ominus (e_1 + 2 e_2 + 3 e_3)$

Out[7]= 6

The following is an example of an inner product of two equal grade elements. An inner product, just as a scalar product, produces a scalar as a result.

In[8]:=
```
{1, 1, 1}.GradeBasis[2]⊖{1, 2, 3}.GradeBasis[2]
% // ToMetricElements
```
Out[8]= $(e_1 \wedge e_2 + e_1 \wedge e_3 + e_2 \wedge e_3) \ominus (e_1 \wedge e_2 + 2 e_1 \wedge e_3 + 3 e_2 \wedge e_3)$

Out[8]= 6

Measure is an inner product of a Grassmann expression with itself.

In[9]:=
```
Measure[3 e₁ ∧ e₂ + 4 e₁ ∧ e₃]
```
Out[9]= 5

An interior product result is zero if the higher grade element is on the right. The interior product is not commutative.

In[10]:=
```
e₂ ⊖ e₂ ∧ e₃
% // ToMetricElements
```
Out[10]= $e_2 \ominus (e_2 \wedge e_3)$

Out[10]= 0

In[11]:=
```
e₂ ∧ e₃ ⊖ e₂
% // ToMetricElements
```
Out[11]= $(e_2 \wedge e_3) \ominus e_2$

Out[11]= e_3

Because ***the interior product is not associative*** the following inputs are *not* equivalent, as can be seen by looking at the output form that shows precedence. The normal precedence is from left to right.

In[12]:=
```
★P
{x⊖y⊖z, x⊖(y⊖z)}
% // ★𝒢
```
Out[12]= $\{(x \ominus y) \ominus z, x \ominus (y \ominus z)\}$

Out[12]= $\{0, x (y \ominus z)\}$

The `ContractInteriorProducts` routine is generally a faster alternative to `ToMetricElements`.

In[13]:=
```
★ℬ₇;
step1 = (e₁ + 2 e₂ + 3 e₃ + 4 e₄ + e₅ + 2 e₆ + e₇) ⊖ (a e₁ + b e₂ + c e₃ + e₄ - e₅ - 3 e₆ + d e₇);
(step1 // ContractInteriorProducts) // Timing
(step1 // ToMetricElements) // Timing
```
Out[13]= {0.124801, -3 + a + 2 b + 3 c + d}

Out[13]= {1.04521, -3 + a + 2 b + 3 c + d}

See Also

ToMetricElements · **ContractInteriorProducts** · **SimplifyInteriorProducts** · **ExpandInteriorProducts** · **ExpandAndSimplifyInteriorProducts** · **ConvertInteriorToInner**

Measure · **ExteriorProduct** · **RegressiveProduct** · **GrassmannComplement**

Related Guides

- Expressions
- Grassmann Calculus

- Interior Products