VAGC_vector_deriv_M_Exercise5.26.wxm

A short development document for Geometric Algebra with wxMaxima
just to test some calculus functions within the GAwxM environment,
contains...
Initialization
Loading of functions (intrinsic and GA specific)
Pseudoscalar definition (specifies the space dimension) and
Calculation of the inverse pseudoscalar used to generate the dual of a multivector
Enumeration of the standard basis for the specified dimension

Exercise 5.26, VAGC page 75 for the derivative of a vector function
on a surface (manifold) in 3D using the (M)anifold functions found in file gcfns3.wxm

Initialization

```
(%i47) ext:["wxm"]$
       file_type_maxima:append(ext,file_type_maxima)$
       batchload("initialize_fns")$
```

the pseudoscalar and its inverse
the lowest useable dimension pseudoscalar should be {e1,e2} i.e. Plen = 2
e.g. for four dimensions edit Pseudos:{e1,e2,e3}$ to Pseudos:{e1,e2,e3,e4}$

```
(%i1) Pseudos:{e1,e2,e3}$
      Pvar:listofvars(Pseudos)$
      Plen:length(Pvar)$
      I:Pseudos$
      ni:(Plen-1)*Plen/2$
      Ii:(-1)^ni*I$
      kill(ni)$
      ldisplay(Pvar)$
```

(%t8) Pvar = [e1, e2, e3]

```
(%i9) batchload("initialize_lsts")$
```

(%t9) lstblds = [[{e1},{e2},{e3}],[{e1,e2},{e1,e3},{e2,e3}],[{e1,e2,e3}]]
(%t10) allblds = [{e1},{e2},{e3},{e1,e2},{e1,e3},{e2,e3},{e1,e2,e3}]
(%t11) invblds = [{e1},{e2},{e3},-{e1,e2},-{e1,e3},-{e2,e3},-{e1,e2,e3}]

end of Initialization

set derivabbrev:false$

```
(%i12) derivabbrev:false$
```

Exercise 5.26
VAGC page 75
repeated from VAGC_vector_deriv_Exercise5.26.wxm

parameterize a surface

```
(%i13) xuv:u*{e1}+v*{e2}+(u*u+v*v)*{e3}$
       ldisplay(xuv)$
```

(%t14) $xuv=\{e3\}*(v^2+u^2)+\{e2\}*v+\{e1\}*u$

find the basis

```
(%i15) xu:diff(xuv,u)$
       xv:diff(xuv,v)$
       ldisplay(xu,xv)$
```

(%t17) $xu=2*\{e3\}*u+\{e1\}$
(%t18) $xv=2*\{e3\}*v+\{e2\}$

find the reciprocal of the basis using Problem 5.4.2

```
(%i19) b2b1:xv&^xu$
       abs2:normod(b2b1)^2$
       xv&*b2b1/abs2$
       b1:facsum(%,allblds)$
```

```
(%i23) b1b2:xu&^xv$
       abs2:normod(b1b2)^2$
       xu&*b1b2/abs2$
       b2:facsum(%,allblds)$
```

```
(%i27) ldisplay(b1,b2)$
```

(%t27) $$b1=\frac{\{e1\}*(4*v^2+1)-4*\{e2\}*u*v+2*\{e3\}*u}{4*v^2+4*u^2+1}$$

(%t28) $$b2=-\frac{4*\{e1\}*u*v-2*\{e3\}*v-\{e2\}*(4*u^2+1)}{4*v^2+4*u^2+1}$$

define a vector function on the surface

```
(%i29) fuv:(v+1)*xu+u*u*xv$
       ldisplay(fuv)$
```

(%t30) $fuv=u^2*(2*\{e3\}*v+\{e2\})+(2*\{e3\}*u+\{e1\})*(v+1)$

form the vector derivative; "vector del" &* "vector f" = bivector + scalar

```
(%i31) b1&*diff(fuv,u)+b2&*diff(fuv,v)$
       delf:facsum(%,allblds)$
       ldisplay(delf)$
```

(%t33) $$delf=(2*\{e1,e3\}*(8*u*v^3+4*v^3+4*v^2-4*u^3*v-4*u^2*v+2*u*v+1)-2*\{e2,e3\}*u*(8*u*v^2+4*v^2+4*v-4*u^3-4*u^2+u-1)+\{e1,e2\}*(8*u*v^2-4*u^2+2*u-1)+4*u*(u*v+v+1))/(4*v^2+4*u^2+1)$$

now using the (M)anifold functions found in file gcfns3.wxm

```
(%i34) bx:u*{e1}+v*{e2}+(u*u+v*v)*{e3}$
       bu:diff(bx,u)$
       bv:diff(bx,v)$
```

```
(%i37) tgtbasis:['bu,'bv]$
       rectgt:reciprocM(tgtbasis)$
       b1:rectgt[1]$
       b2:rectgt[2]$
```

```
(%i41) parlst:[u,v]$
       reclst:['b1,'b2]$
       ldisplay(parlst,reclst)$
```

(%t43) parlst = [u, v]
(%t44) reclst = [b1, b2]

```
(%i45) bf:(v+1)*bu+u*u*bv$
       bfstr:"bf"$
```

```
(%i47) vectordelM(bfstr,parlst,reclst);
```

(%o47) b2&* *diff*(bf,v)+b1&* *diff*(bf,u)

```
(%i48) ev(%)$
       delfM:facsum(%,allblds)$
       ldisplay(delfM)$
```

(%t50) $$delfM=(2*\{e1,e3\}*(8*u*v^3+4*v^3+4*v^2-4*u^3*v-4*u^2*v+2*u*v+1)-2*\{e2,e3\}*u*(8*u*v^2+4*v^2+4*v-4*u^3-4*u^2+u-1)+\{e1,e2\}*(8*u*v^2-4*u^2+2*u-1)+4*u*(u*v+v+1))/(4*v^2+4*u^2+1)$$

show that the two methods are equivalent

```
(%i51) is(equal(delfM,delf));
```

(%o51) true