VAGC_polar_grad.wxm

GNU General Public License 2019 Stephen Athel Abbott.

A development document for Geometric Algebra using wxMaxima

Exercise 5.20, VAGC page 70 for the gradient in polar coordinates for 2D only

Initialization

```
(%i40)  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]$
        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]

(%i9)   batchload("initialize_lsts")$

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

end of Initialization

set derivabbrev:false$

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

Exercise 5.20
VAGC page 70

form the coordinate vector, x from the lists of coefficients in 2D only!

```
(%i13)  xstr:"x"$
        xlst:lstvector(xstr)$
        ldisplay(xlst)$

 (%t15) xlst=[x1,x2,0]

(%i16)  xv:makevector(xlst)$
        ldisplay(xv)$

 (%t17) xv={e2}*x2+{e1}*x1

(%i18)  x1:r*cos(th)$
        x2:r*sin(th)$
        x:ev(xv)$
        ldisplay(x)$

 (%t21) x={e2}*r*sin(th)+{e1}*r*cos(th)
```

only wr is a unit vector of wjbasis

```
(%i22)  wr:diff(x,r)$
        wth:diff(x,th)$
        ldisplay(wr,wth)$

 (%t24) wr={e2}*sin(th)+{e1}*cos(th)
 (%t25) wth={e2}*r*cos(th)-{e1}*r*sin(th)

(%i26)  wjbasis:[wr,wth]$
```

find the reciprocal of the basis with the usage of reciprocb() given in gafns5.wxm

```
(%i27)  bld:list2vecouter(wjbasis)$
        denext:extract(bld)$
        normden:trigsimp(denext[2])$
        wkbasis_numerators:reciprocb(wjbasis)$
        wkr:wkbasis_numerators[1]/normden$
        wkth:wkbasis_numerators[2]/normden$
        ldisplay(wkr,wkth)$

 (%t33)/R/ wkr={e2}*sin(th)+{e1}*cos(th)
 (%t34)/R/ wkth=-(-{e1}*sin(th)-{e2}*cos(th))/r

(%i35)  wkbasis:[wkr,wkth]$
```

Exercise 5.20, part a) use Equation (5.18)

```
(%i36)  gradF:'wkr*'diff(F,r)+'wkth*'diff(F,th)$
        ldisplay(gradF)$

 (%t37) gradF=wkth*('d/d*th*F)+wkr*('d/d*r*F)
```

identify the unit base vectors

```
(%i38)  rhat:wr$
        thetahat:ratsimp(wth/r)$
        ldisplay(rhat,thetahat)$

 (%t40) rhat={e2}*sin(th)+{e1}*cos(th)
 (%t41) thetahat={e2}*cos(th)-{e1}*sin(th)
```

inspection of the equation for gradF

```
(%i42)  is(equal(wkr,rhat));
        is(equal(wkth,thetahat/r));

 (%o42) true
 (%o43) true
```

Exercise 5.20, part b)

find the lhs of part b) as grad(F) for scalar F = f(r,th)

```
(%i44)  F:f(r,th)$
        ev(gradF,diff)$
        lhs:facsum(%,allblds);

 (%o46) -({e1}*(('d/d*th*f(r,th))*sin(th)-r*('d/d*r*f(r,th))*cos(th))-{e2}*
(r*('d/d*r*f(r,th))*sin(th)+('d/d*th*f(r,th))*cos(th)))/r
```

find the rhs of part b) with rhat and thetahat

```
(%i47)  diff(f(r,th),r)*rhat+(1/r)*diff(f(r,th),th)*thetahat$
        rhs:facsum(%,allblds);

 (%o48) -({e1}*(('d/d*th*f(r,th))*sin(th)-r*('d/d*r*f(r,th))*cos(th))-{e2}*
(r*('d/d*r*f(r,th))*sin(th)+('d/d*th*f(r,th))*cos(th)))/r

(%i49)  is(equal(lhs,rhs));

 (%o49) true
```

&. and &^ will need special treatment when F is a vector field

```
(%i50)  F:fr(r,th)*rhat+fth(r,th)*thetahat$
        dFr:diff(F,r);
        dFth:diff(F,th);

 (%o51) ('d/d*r*fr(r,th))*({e2}*sin(th)+{e1}*cos(th))+('d/d*r*fth(r,th))*
({e2}*cos(th)-{e1}*sin(th))

 (%o52) ('d/d*th*fr(r,th))*({e2}*sin(th)+{e1}*cos(th))+fth(r,th)*
(-{e2}*sin(th)-{e1}*cos(th))+('d/d*th*fth(r,th))*({e2}*cos(th)-{e1}*sin(th))+fr(r,th)
*({e2}*cos(th)-{e1}*sin(th))
```

Exercise 5.20, part c)

```
(%i53)  wkr&.dFr+wkth&.dFth$
        expand(%)$
        divF:trigsimp(%)$
        ldisplay(divF)$

 (%t56) divF=(('d/d*th*fth(r,th))+r*('d/d*r*fr(r,th))+fr(r,th))/r
```

differentiation of a product and inspection of the rhs of the equation for divF

```
(%i57)  diff((r*fr(r,th)),r);

 (%o57) r*('d/d*r*fr(r,th))+fr(r,th)
```

Exercise 5.20, part d)

```
(%i58)  wkr&^dFr+wkth&^dFth$
        expand(%)$
        trigsimp(%)$
        curlF:facsum(%,allblds)$
        ldisplay(curlF)$

 (%t62) curlF=({e1,e2}*(r*('d/d*r*fth(r,th))-'d/d*th*fr(r,th)+fth(r,th)))/r
```

show that rhat^thetahat = e1~e2

```
(%i63)  wr&^(wth/r);
        rhat&^thetahat;
        trigsimp(%);

 (%o63)/R/ {e1,e2}*sin(th)^2+{e1,e2}*cos(th)^2
 (%o64)/R/ {e1,e2}*sin(th)^2+{e1,e2}*cos(th)^2
 (%o65) {e1,e2}
```

differentiation of a product and inspection of the rhs of the equation for curlF

```
(%i66)  diff((r*fth(r,th)),r);

 (%o66) r*('d/d*r*fth(r,th))+fth(r,th)
```