

Einar Magnús Einarsson

Myndræn framsetning á gögnum úr reiknilíkaninu MM5

Myndræn framsetning á gögnum úr reiknilíkaninu MM5*

Greinargerð

Einar Magnús Einarsson

febrúar, 2004

1 Um verkefnið

Verkefnið fólst í að koma upp viðmóti sem framreiðir niðurstöður líkanreikninga á veðri og þróun þess. Ætlunin var að viðmótið yrði jafnt fyrir almenna notendur veðurspáa í tengslum við atvinnulíf og frístundir sem og aðila sem stunda ýmiskonar rannsóknir og þróunarvinnu sem byggir á reikningum á veðri. Þá er meðal annars vísað til tölfræðilegrar úrvinnslu veðurþátta, þróunar á síum við veðurspágerð og greiningar á vitlausum spám.

Ákveðið var að notast við IDL forritunarumhverfið við alla myndræna framsetningu á líkanareikningunum þar sem fyrir hendi var tól í IDL til aflestrar á gögnum úr MM5 reiknilíkaninu.

Loks var hönnuð vefsíða sem umgjörð fyrir birtingu á kortum og ritum.

2 Idl

Eins og fyrr segir var notast við IDL hugbúnaðarpakkann til framreiðslu á kortum og ritum. Skrifðar voru 5 rútnur og er hlutverk þeirra eftirfarandi:

- teikna_V.pro: Teiknar vindhraðakort með vigrum, hraði táknaður með stærð vigurs sem og bakgrunnslit.
- teikna_A.pro: Teiknar vindhraðakort með örvum, jafnþrýstilínum og úrkomu.
- teikna_T.pro: Teiknar hitakort með jafnþrýstilínum.
- teikna_L.pro: Teiknar veðurspáritin.
- teikna_skewT.pro: Teiknar hitastig (T) og daggarmark (T_d) sem fall af þrýstingi, $\log(P)$.

*<http://www.mmm.ucar.edu/mm5>

Þessi forrit styðjast svo við nokkur hjálparforrit sem jafnt voru skrifuð beint sem og fengin að „láni“ frá veðurfræðingum sem staðið hafa í svipuðu stappi.

Notast var við MM5IDL¹ pakkan fyrir IDL til aflestrar á MM5 gögnum en sá hugbúnaður var skrifaður af Örnólfi og Ólafi Rögnavaldsonum.

Nánari útlistun á hverju forriti fyrir sig má sjá hér í undiköflunum.

2.1 Veðurkort

Veðurkort eru teiknuð jafnt fyrir allt landið í heild og eins er því skipt upp í níu hluta sem sýna upplausn líkanareikningana betur.

Eins og fyrir segir eru það þrjú forrit sem notuð eru til teikninga á veðurkortum:

- teikna_V.pro
- teikna_A.pro
- teikna_T.pro

Hægt er að stjórna nokkru virkni þeirra fyrir keyrslu eins og sjá má í töflu 1. Það er að segja hægt er að ráða hvaða hlutar landsins eru teiknaðir og í sumum tilvikum hversu mikið á að grisja gögn á landskortu.

Tafla 1: Valmöguleikar teikna_A, teikna_V og teikna_T. GRISJA-valmöguleikinn á þó ekki við teikna_T og BARBS á einungis við teikna_V.

PRO TEIKNA_V,	NIU_1=niu_1,	\$; Vestfirðir
	NIU_2=niu_2,	\$; Norðurland
	NIU_3=niu_3,	\$; Norðausturland
	NIU_4=niu_4,	\$; Vesturland
	NIU_5=niu_5,	\$; Miðhálandið
	NIU_6=niu_6,	\$; Austfirðir
	NIU_7=niu_7,	\$; Suðvesturland
	NIU_8=niu_8,	\$; Suðurland
	NIU_9=niu_9,	\$; Suðausturland
	LAND=land,	\$; Teikna allt landið
	ALLT=allt,	\$; Teikna allt
	GRISJA=grisja,	\$; Grisjun fyrir landið (sjálfvirkt=2)
	BARBS=barbs		; Teikna líka kort með vindörvum.

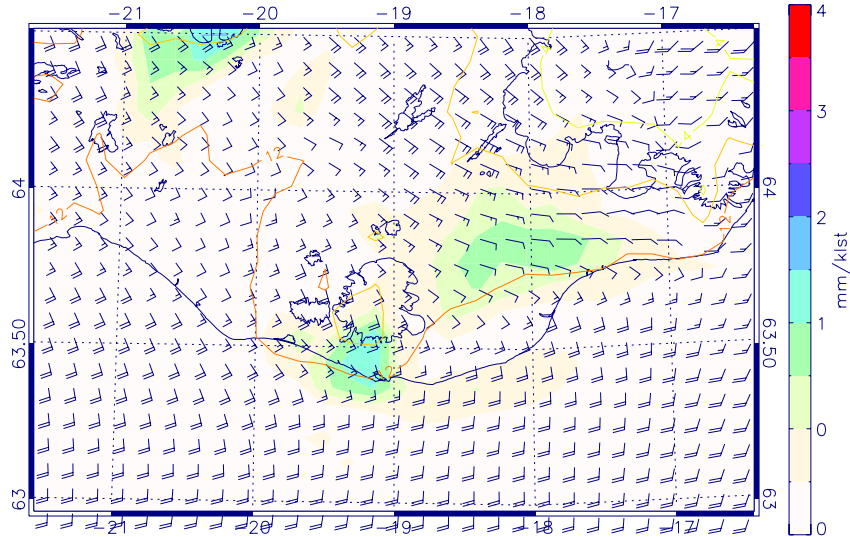
Sauðakóði allra forritanna er nánast eins. Sjá má nokkuð grófa mynd hans í töflu 2.

Útlínur lands, jökla og vatna eru fengnar úr skránum isl.xy, votn.xy, og joklar.xy sem vistaðar eru á Faldi.

Hjálparforritin sem hér eru nýtt eru windbarb.pro sem breytt var til þess að örvar fylgji viðteknum reglum, það er að segja að hálftrístrík á ör merki 2.5 m/s, eitt strík þá 5 m/s

¹<http://www.os.is/~or/mm5idl/>

14.08.03 kl: 18:00



Mynd 1: Dæmi um veðurkort.

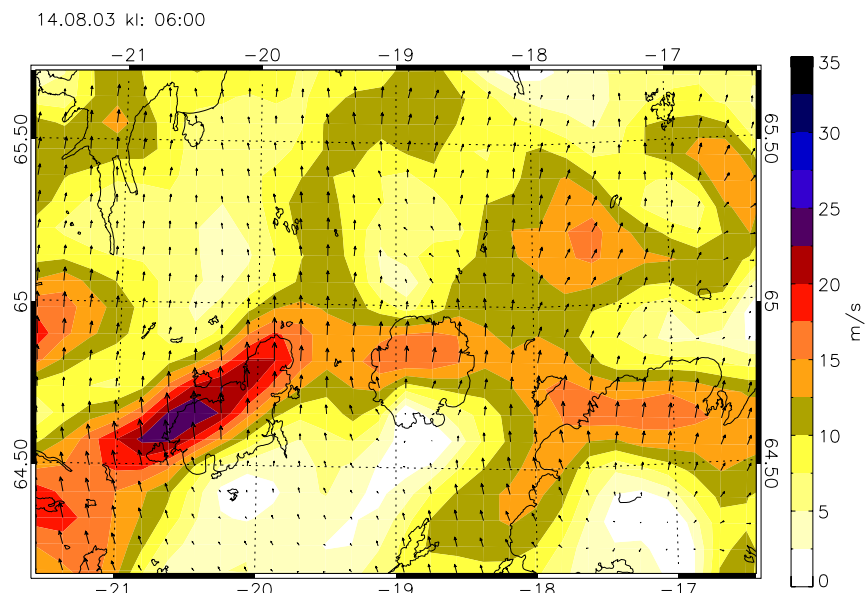
en þrýhyrningur 25 m/s og heitir endurbætta forritið windbarb_eme.pro. Forritinu windbarb.pro var fengið „lánað“ af <http://www.dfanning.com>. Annað forritið er velovect.pro sem kemur af sjálfri IDL kúnni, það teiknar styrk vigursviðs, eða vinds, sem vigur þar sem stærð vigursins er í réttu hlutfalli við vindhraða. Því þurfti að breyta þannig að stærsta örin væri kvörðuð við ákveðna vindhæð, til að kort yrðu samanburðarhæf, og að forritið gæti tekið inn staðsetningu fyrir hvern vigur fyrir sig. Eftir þessar endurbætur heitir forritið mvelovect_or.pro.

Tafla 2: Grófur sauðakóði forritana er að mestu eins.

- > náð í dot/cross punkta úr terfile
- > lesið úr lstfile (listi.lst) þ.e.a.s. hvaða MM5 skrá á að nota
- > tímagildi fengin úr líkani
- > gögn lesin fyrir öll tímagildi (24)
- > möndlað með gögnin (press, td ofl reiknað)
- > réttir punktar fundnir (skewT og spárit) (byggt á poi.pro)
- > gögn sett inn í fylki (skewT og spárit)
- > teiknað uppúr fylkinu (skewT og spárit)
- > landið teiknað (kortagerðarrútínurnar)
- > gögn teiknuð yfir landið (kortagerðarrútínurnar)
- > út kemur postscript skrá.

Þrýstingur við yfirborð kemur ekki beint út úr MM5 líkaninu en til að finna hann er notuð eftirfarandi jafna:

$$P(i, j, sigma, t) = PSTAR(i, j) * sigma(i, j) + PTOP(i, j) + PP(i, j, sigma, t) \quad (1)$$



Mynd 2: Dæmi um vindhraðakort.

Gildi allra breytanna, $PTOP$, $PSTAR$, $sigma$ og PP eru fengin úr líkaninu. Til að hægt sé að plotta þrýsting á kortin sjálf þurfti svo að reikna þrýstinginn „niður í sjávarmál“ en til þess er notað forritið `reduce_to_slp.pro`.

2.2 Spárit

Spárit eru teiknuð fyrir allar mannaðar skeytastöðvar sem og flesta þéttbýliskjarna landsins. Fundnir eru þeir punktar í neti líkansins sem næst liggja hnattstöðu staðanna. Þann lista les forritið inn úr skránni `vedurstod.lst`.

Á hverju spáriti kemur fram vindhraði og vindátt, hitastig og úrkoma á hverri klst. Hugmyndin var að hafa skýjahulu líka en engar notahæfar aðferðir til þess fundust. Í þessu samhengi má einnig benda á það að komið hefur fram sú skoðun að óvön augu gætu átt í erfiðleikum með að lesa vindhæðina af vindörvunum og gæti verið betra að birta hana á línuritaformi.

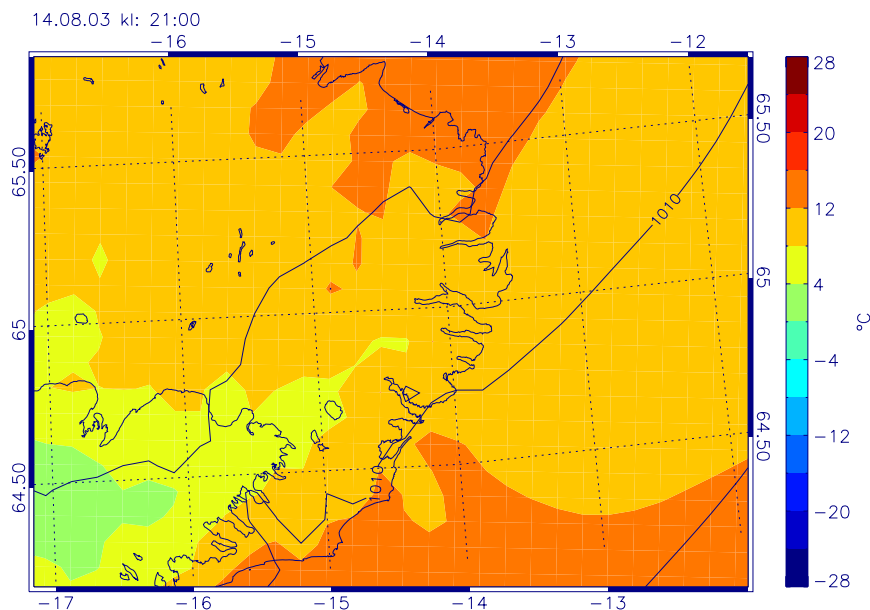
Tekinn eru inn hnattstaða, nafn og fleiri upplýsingar úr skrá, `vedurstod.lst`, af þeim stöðum sem teikna á spárit fyrir. Gerð skrárinnar er eftirfarandi:

```

1 HH.HH HH.HH I          GERÐ NAFN
2 HH MM HH MM I  HEIMILD NAFN
3 HH MM SS HH MM SS I GPSN NAFN
4 HH MM.MMM HH MM.MMM I HEIMILD NAFN
5 NAFN

```

Fyrsta tala línunnar gefur til kynna tegund línu. Næst kemur gerð framsetningar hnattstöðunnar því næst I sem hér er alltaf 0, en þessi tala á að gefa möguleika á, ef vilji er fyrir hendi að brúa inn í möskva líkansins. Gerð segir til um gerð veðurstöðvar, heimild segir til um



Mynd 3: Dæmi um hitakort.

hvaðan gögnin eru fengin, það er að segja ef þau eru ekki beint af veðurstofunni. GPSN er nafn GPS punkts samkvæmt skráningarkerfi Norrænu eldfjallastöðvarinnar. Síðasti reiturinn er síðan ávallt NAFN sem einfaldlega á að vera nafn staðarins, hvort sem um er að ræða veðurstöð, þorp eða áningarstað ferðamanna.

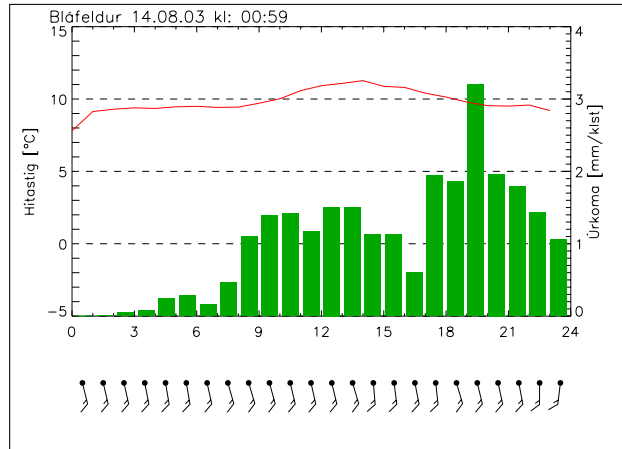
Upplýsingar um hnattstöðu úr skránni voru síðan nýttar til að finna þann punkt úr MM5 netinu er stendur tilsvareandi stað næst.

Tvö hjálparforrit, `wind_barb.pro` og `bar_plot.pro` eru notuð og þörfnuðust bæði smávægilegra breytinga. Því fyrra þurfti að breyta til að hægt væri að nota það fyrir sémilegar postscript skrár og einnig þurfti að kvarða það þannig að örvar sýndu vindhraða í m/s. Endurbætta útgáfan ber nafnið `wind_barb_ps.pro`. Í því seinna var einungis bætt einum valmöguleika í `bar_plot` aðferðina sem fylgir IDL, en bætt var við þeim möguleika að aðferðin skilaði engum ás (`/noaxis`), þannig að útliti og gerð hans er stjórnað með öðrum aðferðum. Endurbætta útgáfan ber nafnið `bar_plot_noaxis.pro`.

Við útlitshönnun var einfaldlega reynt að herma eftir þeim spáritum sem til eru nú þegar. Ásar ritanna eru ekki fastir en hitaás spannar þó alltaf 20°C en hámarksgildi hans ákvarðast þannig að fundið er hæsta hitagildi tímabils og hámarkið sett sem næsta gildi fyrir ofan sem 5 gengur upp í.

Svipaða sögu er að segja um úrkomu ásin, en ef úrkoma er undir 4mm/klst er hann sjálfkrafa látinn spanna 0-4 mm/klst en ef það nægir ekki er hámark ássins sett sem næsta tala fyrir ofan hámarkið sem 5 gengur upp í.

Við gerð spáritanna fengust upplýsingar um gps hnit þeirra staða þar sem ekki eru veðurathugunarstöðvar til staðar af vefsíðu Norrænu eldfjallastöðvarinnar (www.norvol.hi.is).



Mynd 4: Dæmi um spárit.

2.3 Hitarit

Gröf sem sýna daggarmark og hitastig með hæð eru teiknuð fyrir 35 staði sem er jafndreift yfir landið og miðin. Forritið teikna `_skewT.pro` finnur þann hnútpunkt í neti líkansins sem næstur er hnattstöðu Keflavíkurflugvallar og er sá punktur látinn ráða staðsetningu 5×7 punkta netsins. Hægt er að láta forritið teikna gröf fyrir aðra stærð netsins með `net=[y,x]` valmöguleikanum þegar forritið er keyrt, en sjálfgefið er `[5,7]`.

Einnig er á sama grafi teiknaður vindhraði og stefna með hæð. Þau gildi eru sýnd á 100 hPa fresti og þurfti að brúa gögnin á þessa þrýstihæðarfleti.

Daggarmark er reiknað með `td.pro`, sem notar eftirfarandi jöfnur:

$$el = \log \frac{q}{0.62197} * \frac{P/100}{1. + (q/0.62197)} \quad (2)$$

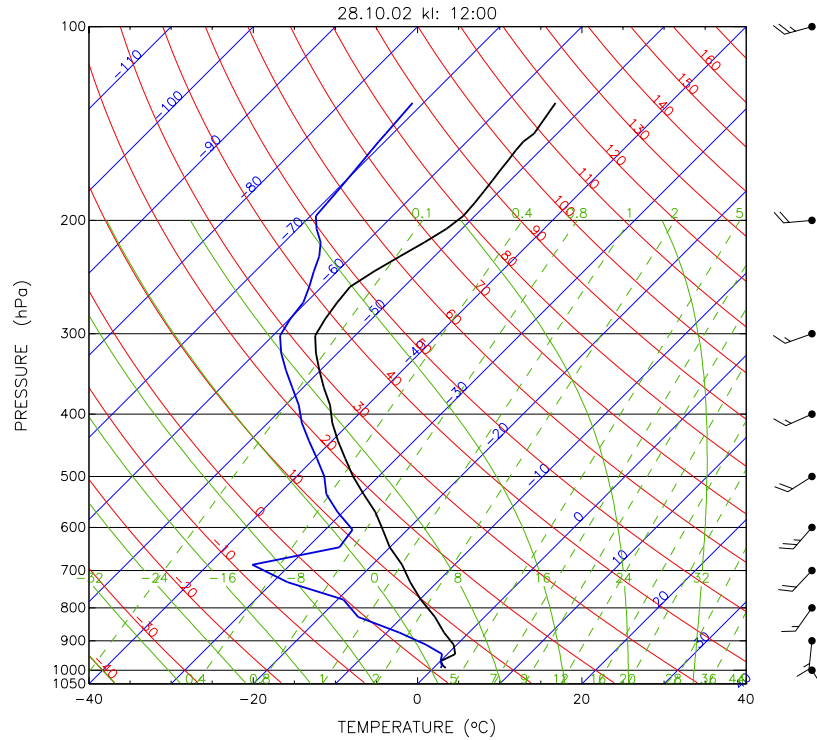
$$t_d = 273.15 + \frac{243.5 * el - 440.8}{19.48 - el} \quad (3)$$

sem fengin er úr fortran forritinu `td.gs` (<http://www.ems.psu.edu/~brian/mm5/grads/td.gs>) sem gert er fyrir MM5 gögn.

Notast er við þrjú hjálparforrit sem öllum þurfti að breyta lítillega. Fyrst er það `skewt.pro` sem teiknar bakgrunn grafsins og setur upp hnitakerfið. Einungis smávægilegri breytingu er þar skotið inn; ein í setning til að litir sjáist í postscript myndum, en annars er keyrsla forritsins óbreytt. Til að teikna hita, daggarmark og vind inn á grafið er forritið `plot_skewt.pro` notað. Því þurfti að breyta til að taka inn gildi fyrir vindhæð og stefnu sem og þrýstingsgildi fyrir vindinn. Einnig þurfti að gæða forritið þeim hæfileika að teikna upp vindörvarnar á tilætlaðan stað, til þess var notast við `wind_barb_ps.pro` eins og í spáritunum.

3 Vefsíða

Vefsíða var hönnuð með það í huga að vera einföld og skýr, ekkert óþarfa þjátur en þó sem einföldust til að hver sem „mús getur valdið“ ætti að geta nýtt sér hana vandræðalaust. Einnig var þess gætt að hún passaði inn í innra net veðurstofunnar.



Mynd 5: Dæmi um hitarit.

Við framsetningu á kortum í tímaröð er notast við Javaskriftur sem hannaðar voru af Philippe Crochet. Einnig var framsetning hans á kortum og gögnum á síðum sem hann hefur umsjón með notuð til hliðjónar hér. En hinsvegar var farinn önnur leið í hönnun síðunnar því að hér eru ekki notaðir rammar (e. frames) til að aðskilja mismunandi fleti síðunnar heldur er notast við CSS.

Til að vefsíðan njóti sín sem best er best að hafa skjá með upplausnina 1280*960 dílar eða betri en enginn vandkvæði eru þó með skjái með lægri upplausn.

A Nauðsynleg forrit

Eftirfarandi forrit eru nauðsynleg til að hægt sé að keyra öll forritin.

- windbarb_eme.pro
- wind_barb_ps.pro
- skew-t.pro
- plot_skewt.pro
- mvelovect_or.pro
- reduce_to_slp.pro
- td.pro
- mm5toIDL hugbúnaðarpakki ÖER og ÓR
- bar_plot_noaxis

Nokkrar hjálpar skeljar; tvær þær fyrri þarfnast þeirra tveggja síðari:

- png_a_net: Breytir kortum og spáritum úr .ps yfir í .png og kemur þeim á réttan stað til birtingar. Býr einnig til .mng hreyfimyndir úr kortunum.
- skewTps2png: Gerir það sama en fyrir skew-T gröfin.
- convert: Breytir .png myndum yfir í .mng hreyfimyndir.
- pstoimg: Breytir .ps skjölum yfir í .png myndir.

B Viðauki

Hér fylgja með IDL kóði þriggja forrita, teikna_A.pro sem er í grunninnn alveg eins og teikna_V.pro og teikna_T.pro, teikna_L.pro og teikna_skewT.pro.

B.1 teikna_A.pro

```
1  @date2timeMM5
2  @pruneMM5
3  @periodMM5
4  @totalMM5__define
5  @colorbar
6  function shft,f
7  gg=f
8  dim = size(gg,/dim)
9  gg=rebin(gg,2*dim[0],2*dim[1])
10 gg=shift(gg,1,1)
11 gg=rebin(gg,dim[0],dim[1])
12 return,gg
13 end
14
15 PRO TEIKNA_A, NIU_1=niu_1, $           ; Vestfirðir
16             NIU_2=niu_2, $           ; Norðurland
17             NIU_3=niu_3, $           ; Norðausturland
18             NIU_4=niu_4, $           ; Vesturland
19             NIU_5=niu_5, $           ; Miðhálandið
20             NIU_6=niu_6, $           ; Austfirðir
21             NIU_7=niu_7, $           ; Suðvesturland
22             NIU_8=niu_8, $           ; Suðurland
23             NIU_9=niu_9, $           ; Suðausturland
24             LAND=land, $             ; Teikna allt landið
25             ALLT=allt, $            ; Teikna allt
26             GRISJA=grisja,$         ; Grisjun fyrir landið (sjálfgefið 5)
27             GRISJA_niu=grisja_niu   ; Landshlutagrisjun (sjálfgefið 1)
28 ;
29 terfile='TERRAIN/TERRAIN_DOMAIN1'
30 ; Let us begin with the mapping stuff
31 mappings,terFile,scalars=terinfo,arrays=terarr
32 ; get the lat/lon values of the dot-points
33 lat_dot = terarr.latitdot
34 lon_dot = terarr.longidot
35 ; now we calculate the i,j (floating) values of the dot-points
36 map2ij,terinfo,lat_dot,lon_dot,idot,jdot
37 ;
38 ; Time to dig into the MM5 data
```

```

39 ;
40 default,lstFile,'listi.lst'
41 b=obj_new('mm5ds',lstFile)
42 b->read
43 dim2d = b->Dim('RAIN NON')
44 dim3d = b->Dim('U')
45 b->SetIterDict,'prune' ; TÍMI
46
47 ; number of timesteps
48 nt = b->nPruned()
49 ; number of sigma levels
50 ns = dim3d[2]
51 p=b->PruneDict()
52
53 ;---úlinur landsins-----
54 utlinur = '/home/sumar1/idl/gogn/faldur/isl.xy'
55 votn = '/home/sumar1/idl/gogn/faldur/votn.xy'
56 joklar = '/home/sumar1/idl/gogn/faldur/joklar.xy'
57 ;---landið-----
58 linufjoldi = File_Lines(utlinur)
59 strond = FltArr(2,linufjoldi-1)
60 haus = 'a'
61 OpenR, lun, utlinur, /Get_Lun
62 ReadF, lun, haus, strond
63 Free_Lun, lun
64 ;--votn-----
65 lin_v = File_Lines(votn)
66 votn_data = StrArr(lin_v)
67 OpenR, lun, votn, /Get_Lun
68 ReadF, lun, votn_data
69 Free_Lun, lun
70 ;--joklar-----
71 lin_j = File_Lines(joklar)
72 joklar_data = StrArr(lin_j)
73 OpenR, lun, joklar, /Get_Lun
74 ReadF, lun, joklar_data
75 Free_Lun, lun
76 ;-----ýmsir fastar-----
77 fjordungar = [[65.25,-18],[64.75,-18],[65.25,-20],[64.75,-20]]
78 xmin = -26
79 xmax = -12
80 ymin = 63
81 ymax = 67
82 x_hlid = 5
83 y_hlid = 1.5

```

```

84
85 topolvlsl=[0.1,250.,500.,750.,1000.,1250.,1500.,1750.]
86 POS  = [0.05,0.05,0.875,0.9]
87 ;-----
88
89 ; Let's get the topography data
90 ter_tmp = b->data(0,'TERRAIN',/array)
91 iinf    = where(finite(ter_tmp, /NaN),cinf)
92 if cinf gt 0 then ter_tmp[iinf]=-0.001
93 ter     = transpose(ter_tmp)
94 ;
95
96 ;-----
97 ;Hér er litavitleysan!
98 ;-----
99 Loadct, 33, Ncolors=14, Bottom=3, /silent
100 levels = 14
101 vlevels= 14
102 rlevels= 8
103 step   = 4
104 mintemp= -28
105 r_levels=Indgen(rlevels)*0.5
106 userLevels = IndGen(levels) * step + mintemp
107 prelevels=findgen(50)*2+950
108 ;-----
109 thisdevice = !D.Name
110 Set_Plot, 'PS'
111 ;-----
112 FOR x=0,nt-2 DO BEGIN
113   u_tmp=b->data(p.keys[x+1], 'U', /array)
114   v_tmp=b->data(p.keys[x+1], 'V', /array)
115   u=transpose(u_tmp)
116   v=transpose(v_tmp)
117   lon=transpose(lon_dot)
118   lat=transpose(lat_dot)
119   u10_tmp=b->data(p.keys[x+1], 'U10', /array)
120   v10_tmp=b->data(p.keys[x+1], 'V10', /array)
121   u10=transpose(shft(U10_tmp))
122   v10=transpose(shft(v10_tmp))
123 ; tímastrengur á mynd
124   dags=strmid(p.keys[x+1],0,16)
125
126   Horn = -!radeg*atan(V10,U10)-90
127   ;Horn = -atan(V,U)*180./!Pi-90.
128   Hradi = sqrt(V10^2.+U10^2.)

```

```

129
130     t2_tmp=b->data(p.keys[x+1], 'T2', /array)
131     T2_kelvin=transpose(t2_tmp)
132     T2=T2_kelvin-273.15
133     fnon0=b->data(p.keys[x], 'RAIN NON', /array)
134     fnon1=b->data(p.keys[x+1], 'RAIN NON', /array)
135     fcon0=b->data(p.keys[x], 'RAIN CON', /array)
136     fcon1=b->data(p.keys[x+1], 'RAIN CON', /array)
137     fcon=fcon1-fcon0
138     fnon=fnon1-fnon0
139     print, mean(fcon1),mean(fcon0),x
140     ; Some cleaning up and unit conversion
141     totprec=10*(fnon+fcon) ;change into mm
142     ; Transpose all the fields so as it fits our way of thinking:-)
143     totprec=transpose(totprec)
144     t_tmp=b->data(p.keys[x+1], 'T', /array)
145     T_kelvin=transpose(t_tmp)
146     T=T_kelvin-273.15
147     ;
148     ; Þrýstingspælingar!
149     ;
150     ptop    = 10000
151     PP_tmp  = b->data(p.keys[x+1], 'PP', /array)
152     PP      = transpose(PP_tmp)
153     PSTAR_tmp = b->data(p.keys[x+1], 'PSTARCRS', /array)
154     PSTAR    = transpose(PSTAR_tmp)
155     sigma_tmp = b->data(p.keys[x+1], 'SIGMAH', /array)
156     sigma     = sigma_tmp
157     Q_tmp    = b->data(p.keys[x+1], 'Q', /array)
158     Q        = transpose(Q_tmp)
159     press=reduce_to_slp((sigma(ns-1)*pstar+pp[ns-1,*,*]+ptop)/100, $
160                      T_kelvin[ns-1,*,*], Q[ns-1,*,*], ter)
161
162
163
164     ;-----LANDID ALLT-----
165     IF N_Elements(land) EQ 1 OR N_Elements(allt) EQ 1 THEN BEGIN
166     ;-----
167     IF N_Elements(grisja) EQ 0 THEN grisja=5
168     U10_grid = congrid(U10,dim2d[1]/grisja,dim2d[0]/grisja)
169     V10_grid = congrid(V10,dim2d[1]/grisja,dim2d[0]/grisja)
170     lon_grid = congrid(lon,dim2d[1]/grisja,dim2d[0]/grisja)
171     lat_grid = congrid(lat,dim2d[1]/grisja,dim2d[0]/grisja)
172     Horn_grid = -!radeg*atan(V10_grid,U10_grid)-90
173     Hradi_grid = sqrt(V10_grid^2.+U10_grid^2.)

```

```

174
175 IF x LT 10 THEN BEGIN
176 Device, filename='A0'+strtrim(string(x),2)+' .ps',/color,Bits_Per_Pixel=8
177 ENDIF ELSE BEGIN
178 Device, filename='A' +strtrim(string(x),2)+' .ps',/color,Bits_Per_Pixel=8
179 ENDELSE
180
181 MAP_SET, 65.,-19.,0, /LAMBERT, LIMIT=[ymin,xmin,ymax,xmax], $
182 POSITION=POS, /clip , /isotropic
183
184 Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
185 contour, totprec[0:dim2d[1]-2,0:dim2d[0]-2], lon[0:dim2d[1]-2,0:dim2d[0]-2],$
186 lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, levels=r_levels, $
187 NLevels=rlevels, /Follow, /cell_fill, C_colors=Reverse(Findgen(rlevels))+1
188
189 Loadct, 0, /silent
190 oplot, strond[0,*], strond[1,*], color=0
191 ; vötn
192 a=IntArr(2)
193 a(0)=0
194 a(1)=strmid(votn_data(0),11,4)
195 FOR s=0,133 DO BEGIN
196 vatn_ll=DblArr(2,a(1))
197 FOR z=a(0),a(0)+a(1)-1 DO BEGIN
198 vatn_ll(0,z-a(0)) = strmid(votn_data(z+1),1,10)
199 vatn_ll(1,z-a(0)) = strmid(votn_data(z+1),12,9)
200 ENDFOR
201 oplot, vatn_ll[0,*], vatn_ll[1,*], color=0
202 a(0)=a(0)+a(1)+1
203 a(1)=strmid(votn_data(a(0)),11,4)
204 ENDFOR
205 ; jökklar
206 a=IntArr(2)
207 a(0)=0
208 a(1)=strmid(joklar_data(0),11,4)
209 FOR s=0,20 DO BEGIN
210 jokl_ll=DblArr(2,a(1))
211 FOR z=a(0),a(0)+a(1)-1 DO BEGIN
212 jokl_ll(0,z-a(0)) = strmid(joklar_data(z+1),1,10)
213 jokl_ll(1,z-a(0)) = strmid(joklar_data(z+1),12,9)
214 ENDFOR
215 oplot, jokl_ll[0,*], jokl_ll[1,*], color=0
216 a(0)=a(0)+a(1)+1
217 a(1)=strmid(joklar_data(a(0)),11,4)
218 ENDFOR

```

```

219 Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
220
221
222
223 colorbar, /vertical, Position = [0.915, 0.05, 0.94, 0.9], $
224 Range=[0,r_levels(rlevels-1)], NColors=rlevels, Bottom=1, /INVERTCOLORS,$
225 Color=!P.color, Divisions=rlevels/2, /right, TITLE='mm/klst'
226
227 Loadct, 0, /silent
228
229 ;contour, press[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
230 ; lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, /FOLLOW, levels=prelevels
231
232
233 windbarb_eme, lon_grid,lat_grid,Hradi_grid,Horn_grid, LENGTH=0.015, $
234 color='black', clipnorm=pos, lim=pos
235
236
237 MAP_GRID, /GRID, LonDel=1, LatDel=0.5, POSISTION=pos, /BOX_AXES, lons=-26
238 ; dags + tími settur á mynd
239 strengur = strmid(dags,8,2)+'.'+strmid(dags,5,2)+'.'+strmid(dags,2,2)+ $
240 ' kl: ' + strmid(dags,11,5)+' '
241 XYOUTS, 0.05, 0.95, strengur, /Normal
242
243 ;HITI
244 Loadct, 33,Ncolors=levels, Bottom=0, /silent
245 contour, T2[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
246 lat[0:dim2d[1]-2,0:dim2d[0]-2], levels=userlevels, $
247 /overplot, /Follow, C_colors=Findgen(levels)
248
249 device,/close_file
250
251 ENDIF ; IF land hættir
252 ;-----01 BYRJAR-----
253 IF N_Elements(niu_1) EQ 1 OR N_Elements(allt) EQ 1 THEN BEGIN
254 ;-----
255 IF N_Elements(grisja_niu) EQ 0 THEN grisja_niu=1
256 U10_grid = congrid(U10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
257 V10_grid = congrid(V10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
258 lon_grid = congrid(lon,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
259 lat_grid = congrid(lat,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
260 Horn_grid = -!radeg*atan(V10_grid,U10_grid)-90
261 Hradi_grid = sqrt(V10_grid^2.+U10_grid^2.)
262 dim2d_grid=size(Hradi_grid, /dim)
263

```

```

264 IF x LT 10 THEN BEGIN
265 Device, filename='01_A0'+strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
266 ENDIF ELSE BEGIN
267 Device, filename='01_A' +strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
268 ENDELSE
269
270 MAP_SET, 65.,-19.,0, /LAMBERT, LIMIT=[ymax-y_hlid,xmin,ymax,xmin+x_hlid], $
271 POSITION=POS, /clip, /isotropic
272
273 Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
274 contour, totprec[0:dim2d[1]-2,0:dim2d[0]-2], lon[0:dim2d[1]-2,0:dim2d[0]-2],$
275 lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, levels=r_levels, $
276 NLevels=rlevels, /Follow, /cell_fill, C_colors=Reverse(Findgen(rlevels))+1
277
278
279 colorbar, /vertical, Position = [0.915, 0.05, 0.94, 0.9], $
280 Range=[0,r_levels(rlevels-1)], NColors=rlevels, Bottom=1, /INVERTCOLORS,$
281 Color=!P.color, Divisions=rlevels/2, /right, TITLE='mm/klst'
282
283
284
285 Loadct, 0, /silent
286 oplot, strond[0,*], strond[1,*], color=0
287 ; vötn
288 a=IntArr(2)
289 a(0)=0
290 a(1)=strmid(votn_data(0),11,4)
291 FOR s=0,133 DO BEGIN
292 vatn_ll=DblArr(2,a(1))
293 FOR z=a(0),a(0)+a(1)-1 DO BEGIN
294 vatn_ll(0,z-a(0)) = strmid(votn_data(z+1),1,10)
295 vatn_ll(1,z-a(0)) = strmid(votn_data(z+1),12,9)
296 ENDFOR
297 oplot, vatn_ll[0,*], vatn_ll[1,*], color=0
298 a(0)=a(0)+a(1)+1
299 a(1)=strmid(votn_data(a(0)),11,4)
300 ENDFOR
301 ; jöklar
302 a=IntArr(2)
303 a(0)=0
304 a(1)=strmid(joklar_data(0),11,4)
305 FOR s=0,20 DO BEGIN
306 jokl_ll=DblArr(2,a(1))
307 FOR z=a(0),a(0)+a(1)-1 DO BEGIN
308 jokl_ll(0,z-a(0)) = strmid(joklar_data(z+1),1,10)

```



```

309  jokl_ll(1,z-a(0)) = strmid(joklar_data(z+1),12,9)
310  ENDFOR
311  oplot, jokl_ll[0,*], jokl_ll[1,*], color=0
312  a(0)=a(0)+a(1)+1
313  a(1)=strmid(joklar_data(a(0)),11,4)
314  ENDFOR
315  windbarb_eme, lon_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
316    lat_grid[0:dim2d[1]-2,0:dim2d[0]-2],$
317    Hradi_grid[0:dim2d[1]-2,0:dim2d[0]-2],$
318    Horn_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
319    LENGTH=0.02, color='black', clipnorm=pos, lim=pos
320
321  ;contour, press[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
322  ;  lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, /FOLLOW, levels=prelevels
323
324  MAP_GRID, /GRID, LonDel=1, LatDel=0.5, POSISTION=pos, /BOX_AXES, lons=-26
325  ; dags + tími settur á mynd
326  strengur = strmid(dags,8,2)+'.'+strmid(dags,5,2)+'.'+strmid(dags,2,2)+ $
327    ' kl: ' + strmid(dags,11,5)+' '
328  XYOUTS, 0.05, 0.95, strengur, /Normal
329
330  ;HITI
331  Loadct, 33,Ncolors=levels, Bottom=0, /silent
332  contour, T2[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
333    lat[0:dim2d[1]-2,0:dim2d[0]-2], levels=userlevels, $
334    /overplot, /Follow, C_colors=Findgen(levels)
335
336  device,/close_file
337
338  ENDIF
339  ;-----02 BYRJAR-----
340  IF N_Elements(niu_2) EQ 1 OR N_Elements(allt) EQ 1 THEN BEGIN
341  ;-----
342  IF N_Elements(grisja_niu) EQ 0 THEN grisja_niu=1
343  U10_grid = congrid(U10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
344  V10_grid = congrid(V10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
345  lon_grid = congrid(lon,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
346  lat_grid = congrid(lat,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
347  Horn_grid = -!radeg*atan(V10_grid,U10_grid)-90
348  Hradi_grid = sqrt(V10_grid^2.+U10_grid^2.)
349  dim2d_grid=size(Hradi_grid, /dim)
350
351  IF x LT 10 THEN BEGIN
352  Device, filename='02_A0'+strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
353  ENDIF ELSE BEGIN

```

```

354 Device, filename='02_A' +strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
355 ENDELSE
356
357 MAP_SET, 65.,-19.,0, /LAMBERT, $
358   LIMIT=[ymax-y_hlid,xmin+x_hlid-0.5,ymax,xmin+2*x_hlid-0.5], $
359   POSITION=POS, /clip , /isotropic
360
361 Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
362 contour, totprec[0:dim2d[1]-2,0:dim2d[0]-2], lon[0:dim2d[1]-2,0:dim2d[0]-2],$
363   lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, levels=r_levels, $
364   NLevels=rlevels, /Follow, /cell_fill, C_colors=Reverse(Findgen(rlevels))+1
365
366 colorbar, /vertical, Position = [0.915, 0.05, 0.94, 0.9], $
367 Range=[0,r_levels(rlevels-1)], NColors=rlevels, Bottom=1, /INVERTCOLORS,$
368 Color=!P.color, Divisions=rlevels/2, /right, TITLE='mm/klst'
369
370
371 Loadct, 0, /silent
372
373 oplot, strond[0,*], strond[1,*], color=0
374 ; vötn
375 a=IntArr(2)
376 a(0)=0
377 a(1)=strmid(votn_data(0),11,4)
378 FOR s=0,133 DO BEGIN
379   vatn_ll=DblArr(2,a(1))
380   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
381     vatn_ll(0,z-a(0)) = strmid(votn_data(z+1),1,10)
382     vatn_ll(1,z-a(0)) = strmid(votn_data(z+1),12,9)
383   ENDFOR
384   oplot, vatn_ll[0,*], vatn_ll[1,*], color=0
385   a(0)=a(0)+a(1)+1
386   a(1)=strmid(votn_data(a(0)),11,4)
387   ENDFOR
388 ; jöklar
389 a=IntArr(2)
390 a(0)=0
391 a(1)=strmid(joklar_data(0),11,4)
392 FOR s=0,20 DO BEGIN
393   jokl_ll=DblArr(2,a(1))
394   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
395     jokl_ll(0,z-a(0)) = strmid(joklar_data(z+1),1,10)
396     jokl_ll(1,z-a(0)) = strmid(joklar_data(z+1),12,9)
397   ENDFOR
398   oplot, jokl_ll[0,*], jokl_ll[1,*], color=0

```

```

399 a(0)=a(0)+a(1)+1
400 a(1)=strmid(joklar_data(a(0)),11,4)
401 ENDFOR
402 windbarb_eme, lon_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
403   lat_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
404   Hradi_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
405   Horn_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
406   LENGTH=0.02, color='black', clipnorm=pos, lim=pos
407
408 ;contour, press[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
409 ; lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, /FOLLOW, levels=prelevels
410
411
412 MAP_GRID, /GRID, LonDel=1, LatDel=0.5, POSISTION=pos, /BOX_AXES, lons=-26
413 ; dags + tími settur á mynd
414 strengur = strmid(dags,8,2)+'.'+strmid(dags,5,2)+'.'+strmid(dags,2,2)+ $
415   ' kl: ' + strmid(dags,11,5)+' '
416 XYOUTS, 0.05, 0.95, strengur, /Normal
417
418
419 ;HITI
420 Loadct, 33,Ncolors=levels, Bottom=0, /silent
421 contour, T2[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
422   lat[0:dim2d[1]-2,0:dim2d[0]-2], levels=userlevels, $
423   /overplot, /Follow, C_colors=Findgen(levels)
424
425 device,/close_file
426 abba=size(V)
427
428 ENDIF
429 ;-----03 BYRJAR-----
430 IF N_Elements(niu_3) EQ 1 OR N_Elements(allt) EQ 1 THEN BEGIN
431 ;-----
432 IF N_Elements(grisja_niu) EQ 0 THEN grisja_niu=1
433 U10_grid = congrid(U10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
434 V10_grid = congrid(V10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
435 lon_grid = congrid(lon,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
436 lat_grid = congrid(lat,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
437 Horn_grid = -!radeg*atan(V10_grid,U10_grid)-90
438 Hradi_grid = sqrt(V10_grid^2.+U10_grid^2.)
439 dim2d_grid=size(Hradi_grid, /dim)
440
441 IF x LT 10 THEN BEGIN
442 Device, filename='03_A0'+strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
443 ENDIF ELSE BEGIN

```

```

444 Device, filename='03_A' +strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
445 ENDELSE
446
447 MAP_SET, 65.,-19.,0, /LAMBERT, LIMIT=[ymax-y_hlid,xmax-x_hlid,ymax,xmax], $
448 POSITION=POS, /clip , /isotropic
449
450 Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
451 contour, totprec[0:dim2d[1]-2,0:dim2d[0]-2], lon[0:dim2d[1]-2,0:dim2d[0]-2],$
452 lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, levels=r_levels, $
453 NLevels=rlevels, /Follow, /cell_fill, C_colors=Reverse(Findgen(rlevels))+1
454
455 colorbar, /vertical, Position = [0.915, 0.05, 0.94, 0.9], $
456 Range=[0,r_levels(rlevels-1)], NColors=rlevels, Bottom=1, /INVERTCOLORS,$
457 Color=!P.color, Divisions=rlevels/2, /right, TITLE='mm/klst'
458
459
460 Loadct, 0, /silent
461
462 oplot, strond[0,*], strond[1,*], color=0
463 ; vötn
464 a=IntArr(2)
465 a(0)=0
466 a(1)=strmid(votn_data(0),11,4)
467 FOR s=0,133 DO BEGIN
468 vatn_ll=DblArr(2,a(1))
469 FOR z=a(0),a(0)+a(1)-1 DO BEGIN
470 vatn_ll(0,z-a(0)) = strmid(votn_data(z+1),1,10)
471 vatn_ll(1,z-a(0)) = strmid(votn_data(z+1),12,9)
472 ENDFOR
473 oplot, vatn_ll[0,*], vatn_ll[1,*], color=0
474 a(0)=a(0)+a(1)+1
475 a(1)=strmid(votn_data(a(0)),11,4)
476 ENDFOR
477 ; jöklar
478 a=IntArr(2)
479 a(0)=0
480 a(1)=strmid(joklar_data(0),11,4)
481 FOR s=0,20 DO BEGIN
482 jokl_ll=DblArr(2,a(1))
483 FOR z=a(0),a(0)+a(1)-1 DO BEGIN
484 jokl_ll(0,z-a(0)) = strmid(joklar_data(z+1),1,10)
485 jokl_ll(1,z-a(0)) = strmid(joklar_data(z+1),12,9)
486 ENDFOR
487 oplot, jokl_ll[0,*], jokl_ll[1,*], color=0
488 a(0)=a(0)+a(1)+1

```

```

489 a(1)=strmid(joklar_data(a(0)),11,4)
490 ENDFOR
491 windbarb_eme, lon_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
492   lat_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
493   Hradi_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
494   Horn_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
495   LENGTH=0.02, color='black', clipnorm=pos, lim=pos
496
497 ;contour, press[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
498 ;   lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, /FOLLOW, levels=prelevels
499
500
501 MAP_GRID, /GRID, LonDel=1, LatDel=0.5, POSISTION=pos, /BOX_AXES, lons=-26
502 ; dags + tími settur á mynd
503 strengur = strmid(dags,8,2)+'.'+strmid(dags,5,2)+'.'+strmid(dags,2,2)+ $
504   ' kl: ' + strmid(dags,11,5)+' '
505 XYOUTS, 0.05, 0.95, strengur, /Normal
506
507 ;HITI
508 Loadct, 33,Ncolors=levels, Bottom=0, /silent
509 contour, T2[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
510   lat[0:dim2d[1]-2,0:dim2d[0]-2], levels=userlevels, $
511   /overplot, /Follow, C_colors=Findgen(levels)
512
513 device,/close_file
514 abba=size(V)
515
516 ENDIF
517 ;-----04 BYRJAR-----
518 IF N_Elements(niu_4) EQ 1 OR N_Elements(allt) EQ 1 THEN BEGIN
519 ;-----
520 IF N_Elements(grisja_niu) EQ 0 THEN grisja_niu=1
521 U10_grid = congrid(U10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
522 V10_grid = congrid(V10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
523 lon_grid = congrid(lon,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
524 lat_grid = congrid(lat,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
525 Horn_grid = -!radeg*atan(V10_grid,U10_grid)-90
526 Hradi_grid = sqrt(V10_grid^2.+U10_grid^2.)
527 dim2d_grid=size(Hradi_grid, /dim)
528
529 IF x LT 10 THEN BEGIN
530 Device, filename='04_A0'+strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
531 ENDIF ELSE BEGIN
532 Device, filename='04_A' +strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
533 ENDELSE

```

```

534
535 MAP_SET, 65.,-19.,0, /LAMBERT, $
536   LIMIT=[ymin+y_hlid-0.3,xmin,ymin+2*y_hlid-0.3,xmin+x_hlid], $
537   POSITION=POS, /clip , /isotropic
538
539
540 Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
541 contour, totprec[0:dim2d[1]-2,0:dim2d[0]-2], lon[0:dim2d[1]-2,0:dim2d[0]-2],$
542   lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, levels=r_levels, $
543   NLevels=rlevels, /Follow, /cell_fill, C_colors=Reverse(Findgen(rlevels))+1
544
545 colorbar, /vertical, Position = [0.915, 0.05, 0.94, 0.9],NColors=rlevels, $
546   /INVERTCOLORS, Divisions=rlevels/2
547 ;, $
548 ;Range=[0,r_levels(rlevels-1)], NColors=rlevels, Bottom=0, /INVERTCOLORS,$
549 ;Color=!P.color, Divisions=rlevels/2, /right, TITLE='mm/klst'
550 ;print, rlevels
551
552 Loadct, 0, /silent
553
554 oplot, strond[0,*], strond[1,*], color=0
555 ; vötn
556 a=IntArr(2)
557 a(0)=0
558 a(1)=strmid(votn_data(0),11,4)
559 FOR s=0,133 DO BEGIN
560   vatn_ll=DblArr(2,a(1))
561   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
562     vatn_ll(0,z-a(0)) = strmid(votn_data(z+1),1,10)
563     vatn_ll(1,z-a(0)) = strmid(votn_data(z+1),12,9)
564   ENDFOR
565 oplot, vatn_ll[0,*], vatn_ll[1,*], color=0
566 a(0)=a(0)+a(1)+1
567 a(1)=strmid(votn_data(a(0)),11,4)
568 ENDFOR
569 ; jöklar
570 a=IntArr(2)
571 a(0)=0
572 a(1)=strmid(joklar_data(0),11,4)
573 FOR s=0,20 DO BEGIN
574   jokl_ll=DblArr(2,a(1))
575   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
576     jokl_ll(0,z-a(0)) = strmid(joklar_data(z+1),1,10)
577     jokl_ll(1,z-a(0)) = strmid(joklar_data(z+1),12,9)
578   ENDFOR

```

```

579 oplot, jokl_ll[0,*], jokl_ll[1,*], color=0
580 a(0)=a(0)+a(1)+1
581 a(1)=strmid(joklar_data(a(0)),11,4)
582 ENDFOR
583 windbarb_eme, lon_grid[0:dim2d_grid[0]-2,0:dim2d_grid[1]-2], $
584   lat_grid[0:dim2d_grid[0]-2,0:dim2d_grid[1]-2], $
585   Hradi_grid[0:dim2d_grid[0]-2,0:dim2d_grid[1]-2], $
586   Horn_grid[0:dim2d_grid[0]-2,0:dim2d_grid[1]-2], $
587   LENGTH=0.02, color='black', clipnorm=pos, lim=pos
588
589 ;contour, press[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
590 ;   lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, /FOLLOW, levels=prelevels
591
592
593 MAP_GRID, /GRID, LonDel=1, LatDel=0.5, POSISTION=pos, /BOX_AXES, lons=-26
594 ; dags + tími settur á mynd
595 strengur = strmid(dags,8,2)+'.'+strmid(dags,5,2)+'.'+strmid(dags,2,2)+ $
596   ' kl: ' + strmid(dags,11,5)+' '
597 XYOUTS, 0.05, 0.95, strengur, /Normal
598
599 ;HITI
600 Loadct, 33,Ncolors=levels, Bottom=0, /silent
601 contour, T2[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
602   lat[0:dim2d[1]-2,0:dim2d[0]-2], levels=userlevels, $
603   /overplot, /Follow, C_colors=Findgen(levels)
604
605 device,/close_file
606 abba=size(V)
607
608 ENDIF
609 ;-----05 BYRJAR-----
610 IF N_Elements(niu_5) EQ 1 OR N_Elements(allt) EQ 1 THEN BEGIN
611 ;-----
612 IF N_Elements(grisja_niu) EQ 0 THEN grisja_niu=1
613 U10_grid = congrid(U10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
614 V10_grid = congrid(V10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
615 lon_grid = congrid(lon,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
616 lat_grid = congrid(lat,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
617 Horn_grid = -!radeg*atan(V10_grid,U10_grid)-90
618 Hradi_grid = sqrt(V10_grid^2.+U10_grid^2.)
619 dim2d_grid=size(Hradi_grid, /dim)
620
621 IF x LT 10 THEN BEGIN
622 Device, filename='05_A0'+strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
623 ENDIF ELSE BEGIN

```

```

624 Device, filename='05_A' +strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
625 ENDELSE
626
627 MAP_SET, 65.,-19.,0, /LAMBERT, $
628   LIMIT=[ymin+y_hlid-0.3,xmin+x_hlid-0.5,ymin+2*y_hlid-0.3,xmin+2*x_hlid-0.5], $
629   POSITION=POS, /clip , /isotropic
630
631
632 Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
633 contour, totprec[0:dim2d[1]-2,0:dim2d[0]-2], lon[0:dim2d[1]-2,0:dim2d[0]-2],$
634   lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, levels=r_levels, $
635   NLevels=rlevels, /Follow, /cell_fill, C_colors=Reverse(Findgen(rlevels))+1
636
637
638 colorbar, /vertical, Position = [0.915, 0.05, 0.94, 0.9], $
639 Range=[0,r_levels(rlevels-1)], NColors=rlevels, Bottom=1, /INVERTCOLORS,$
640 Color=!P.color, Divisions=rlevels/2, /right, TITLE='mm/klst'
641
642
643 Loadct, 0, /silent
644
645 oplot, strond[0,*], strond[1,*], color=0
646 ; vötn
647 a=IntArr(2)
648 a(0)=0
649 a(1)=strmid(votn_data(0),11,4)
650 FOR s=0,133 DO BEGIN
651   vatn_ll=DblArr(2,a(1))
652   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
653     vatn_ll(0,z-a(0)) = strmid(votn_data(z+1),1,10)
654     vatn_ll(1,z-a(0)) = strmid(votn_data(z+1),12,9)
655   ENDFOR
656   oplot, vatn_ll[0,*], vatn_ll[1,*], color=0
657   a(0)=a(0)+a(1)+1
658   a(1)=strmid(votn_data(a(0)),11,4)
659   ENDFOR
660 ; jöklar
661 a=IntArr(2)
662 a(0)=0
663 a(1)=strmid(joklar_data(0),11,4)
664 FOR s=0,20 DO BEGIN
665   jokl_ll=DblArr(2,a(1))
666   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
667     jokl_ll(0,z-a(0)) = strmid(joklar_data(z+1),1,10)
668     jokl_ll(1,z-a(0)) = strmid(joklar_data(z+1),12,9)

```



```

669 ENDFOR
670 oplot, jokl_ll[0,*], jokl_ll[1,*], color=0
671 a(0)=a(0)+a(1)+1
672 a(1)=strmid(joklar_data(a(0)),11,4)
673 ENDFOR
674 Loadct, 0, /silent
675 windbarb_eme, lon_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
676   lat_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
677   Hradi_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
678   Horn_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
679   LENGTH=0.02, color='black', clipnorm=pos, lim=pos
680
681 ;contour, press[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
682 ; lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, /FOLLOW, levels=prelevels
683
684
685 MAP_GRID, /GRID, LonDel=1, LatDel=0.5, POSISTION=pos, /BOX_AXES, lons=-26
686 ; dags + tími settur á mynd
687 strengur = strmid(dags,8,2)+'.'+strmid(dags,5,2)+'.'+strmid(dags,2,2)+ $
688   ' kl: ' + strmid(dags,11,5)+' '
689 XYOUTS, 0.05, 0.95, strengur, /Normal
690
691 ;HITI
692 Loadct, 33,Ncolors=levels, Bottom=0, /silent
693 contour, T2[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
694   lat[0:dim2d[1]-2,0:dim2d[0]-2], levels=userlevels, $
695   /overplot, /Follow, C_colors=Findgen(levels)
696
697 device,/close_file
698 abba=size(V)
699
700 ENDIF
701 ;-----06 BYRJAR-----
702 IF N_Elements(niu_6) EQ 1 OR N_Elements(allt) EQ 1 THEN BEGIN
703 ;-----
704 IF N_Elements(grisja_niu) EQ 0 THEN grisja_niu=1
705 U10_grid = congrid(U10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
706 V10_grid = congrid(V10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
707 lon_grid = congrid(lon,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
708 lat_grid = congrid(lat,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
709 Horn_grid = -!radeg*atan(V10_grid,U10_grid)-90
710 Hradi_grid = sqrt(V10_grid^2.+U10_grid^2.)
711 dim2d_grid=size(Hradi_grid, /dim)
712
713 IF x LT 10 THEN BEGIN

```

```

714 Device, filename='06_A0'+strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
715 ENDEF ELSE BEGIN
716 Device, filename='06_A' +strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
717 ENDELSE
718
719 MAP_SET, 65.,-19.,0, /LAMBERT, $
720   LIMIT=[ymin+y_hlid-0.3,xmax-x_hlid,ymin+2*y_hlid-0.3,xmax], $
721   POSITION=POS, /clip , /isotropic
722
723 Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
724 contour, totprec[0:dim2d[1]-2,0:dim2d[0]-2], lon[0:dim2d[1]-2,0:dim2d[0]-2],$
725   lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, levels=r_levels, $
726   NLevels=rlevels, /Follow, /cell_fill, C_colors=Reverse(Findgen(rlevels))+1
727
728
729 colorbar, /vertical, Position = [0.915, 0.05, 0.94, 0.9], $
730 Range=[0,r_levels(rlevels-1)], NColors=rlevels, Bottom=1, /INVERTCOLORS,$
731 Color=!P.color, Divisions=rlevels/2, /right, TITLE='mm/klst'
732
733
734 Loadct, 0, /silent
735
736 oplot, strond[0,*], strond[1,*], color=0
737 ; vötn
738 a=IntArr(2)
739 a(0)=0
740 a(1)=strmid(votn_data(0),11,4)
741 FOR s=0,133 DO BEGIN
742   vatn_ll=DblArr(2,a(1))
743   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
744     vatn_ll(0,z-a(0)) = strmid(votn_data(z+1),1,10)
745     vatn_ll(1,z-a(0)) = strmid(votn_data(z+1),12,9)
746   ENDFOR
747   oplot, vatn_ll[0,*], vatn_ll[1,*], color=0
748   a(0)=a(0)+a(1)+1
749   a(1)=strmid(votn_data(a(0)),11,4)
750 ENDFOR
751 ; jöklar
752 a=IntArr(2)
753 a(0)=0
754 a(1)=strmid(joklar_data(0),11,4)
755 FOR s=0,20 DO BEGIN
756   jokl_ll=DblArr(2,a(1))
757   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
758     jokl_ll(0,z-a(0)) = strmid(joklar_data(z+1),1,10)

```

```

759  jokl_ll(1,z-a(0)) = strmid(joklar_data(z+1),12,9)
760  ENDFOR
761  oplot, jokl_ll[0,*], jokl_ll[1,*], color=0
762  a(0)=a(0)+a(1)+1
763  a(1)=strmid(joklar_data(a(0)),11,4)
764  ENDFOR
765  windbarb_eme, lon_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
766    lat_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
767    Hradi_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
768    Horn_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
769    LENGTH=0.02, color='black', clipnorm=pos, lim=pos
770
771  ;contour, press[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
772  ; lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, /FOLLOW, levels=prelevels
773
774  MAP_GRID, /GRID, LonDel=1, LatDel=0.5, POSISTION=pos, /BOX_AXES, lons=-26
775  ; dags + tími settur á mynd
776  strengur = strmid(dags,8,2)+'.'+strmid(dags,5,2)+'.'+strmid(dags,2,2)+ $
777    ' kl: ' + strmid(dags,11,5)+' '
778  XYOUTS, 0.05, 0.95, strengur, /Normal
779
780  ;HITI
781  Loadct, 33,Ncolors=levels, Bottom=0, /silent
782  contour, T2[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
783    lat[0:dim2d[1]-2,0:dim2d[0]-2], levels=userlevels, $
784    /overplot, /Follow, C_colors=Findgen(levels)
785
786  device,/close_file
787  abba=size(V)
788
789  ENDIF
790  ;-----07 BYRJAR-----
791  IF N_Elements(niu_7) EQ 1 OR N_Elements(allt) EQ 1 THEN BEGIN
792  ;-----
793  IF N_Elements(grisja_niu) EQ 0 THEN grisja_niu=1
794  U10_grid = congrid(U10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
795  V10_grid = congrid(V10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
796  lon_grid = congrid(lon,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
797  lat_grid = congrid(lat,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
798  Horn_grid = -!radeg*atan(V10_grid,U10_grid)-90
799  Hradi_grid = sqrt(V10_grid^2.+U10_grid^2.)
800  dim2d_grid=size(Hradi_grid, /dim)
801
802  IF x LT 10 THEN BEGIN
803  Device, filename='07_A0'+strtrim(string(x),2)+''.ps',/color,Bits_Per_Pixel=8

```

```

804   ENDIF ELSE BEGIN
805   Device, filename='07_A' +strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
806   ENDELSE
807
808   MAP_SET, 65.,-19.,0, /LAMBERT, LIMIT=[ymin,xmin,ymin+y_hlid,xmin+x_hlid], $
809     POSITION=POS, /clip , /isotropic
810
811
812   Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
813   contour, totprec[0:dim2d[1]-2,0:dim2d[0]-2], lon[0:dim2d[1]-2,0:dim2d[0]-2],$
814     lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, levels=r_levels, $
815     NLevels=rlevels, /Follow, /cell_fill, C_colors=Reverse(Findgen(rlevels))+1
816
817
818   colorbar, /vertical, Position = [0.915, 0.05, 0.94, 0.9], $
819   Range=[0,r_levels(rlevels-1)], NColors=rlevels, Bottom=1, /INVERTCOLORS,$
820   Color=!P.color, Divisions=rlevels/2, /right, TITLE='mm/klst'
821
822
823   Loadct, 0, /silent
824
825   oplot, strond[0,*], strond[1,*], color=0
826   ; vötn
827   a=IntArr(2)
828   a(0)=0
829   a(1)=strmid(votn_data(0),11,4)
830   FOR s=0,133 DO BEGIN
831     vatn_ll=DblArr(2,a(1))
832     FOR z=a(0),a(0)+a(1)-1 DO BEGIN
833       vatn_ll(0,z-a(0)) = strmid(votn_data(z+1),1,10)
834       vatn_ll(1,z-a(0)) = strmid(votn_data(z+1),12,9)
835     ENDFOR
836     oplot, vatn_ll[0,*], vatn_ll[1,*], color=0
837     a(0)=a(0)+a(1)+1
838     a(1)=strmid(votn_data(a(0)),11,4)
839   ENDFOR
840   ; jöklar
841   a=IntArr(2)
842   a(0)=0
843   a(1)=strmid(joklar_data(0),11,4)
844   FOR s=0,20 DO BEGIN
845     jokl_ll=DblArr(2,a(1))
846     FOR z=a(0),a(0)+a(1)-1 DO BEGIN
847       jokl_ll(0,z-a(0)) = strmid(joklar_data(z+1),1,10)
848       jokl_ll(1,z-a(0)) = strmid(joklar_data(z+1),12,9)

```

```

849 ENDFOR
850 oplot, jokl_ll[0,*], jokl_ll[1,*], color=0
851 a(0)=a(0)+a(1)+1
852 a(1)=strmid(joklar_data(a(0)),11,4)
853 ENDFOR
854 windbarb_eme, lon_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
855   lat_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
856   Hradi_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
857   Horn_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
858   LENGTH=0.02, color='black', clipnorm=pos, lim=pos
859
860 ;contour, press[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
861 ; lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, /FOLLOW, levels=prelevels
862
863 MAP_GRID, /GRID, LonDel=1, LatDel=0.5, POSISTION=pos, /BOX_AXES, lons=-26
864 ; dags + tími settur á mynd
865 strengur = strmid(dags,8,2)+'.'+strmid(dags,5,2)+'.'+strmid(dags,2,2)+ $
866   ' kl: ' + strmid(dags,11,5)+' '
867 XYOUTS, 0.05, 0.95, strengur, /Normal
868
869 ;HITI
870 Loadct, 33,Ncolors=levels, Bottom=0, /silent
871 contour, T2[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
872   lat[0:dim2d[1]-2,0:dim2d[0]-2], levels=userlevels, $
873   /overplot, /Follow, C_colors=Findgen(levels)
874
875 device,/close_file
876 abba=size(V)
877
878 ENDIF
879 ;-----08 BYRJAR-----
880 IF N_Elements(niu_8) EQ 1 OR N_Elements(allt) EQ 1 THEN BEGIN
881 ;-----
882 IF N_Elements(grisja_niu) EQ 0 THEN grisja_niu=1
883 U10_grid = congrid(U10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
884 V10_grid = congrid(V10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
885 lon_grid = congrid(lon,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
886 lat_grid = congrid(lat,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
887 Horn_grid = -!radeg*atan(V10_grid,U10_grid)-90
888 Hradi_grid = sqrt(V10_grid^2.+U10_grid^2.)
889 dim2d_grid=size(Hradi_grid, /dim)
890
891 IF x LT 10 THEN BEGIN
892 Device, filename='08_A0'+strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
893 ENDIF ELSE BEGIN

```

```

894 Device, filename='08_A' +strtrim(string(x),2)+'.ps',/color,Bits_Per_Pixel=8
895 ENDELSE
896
897 MAP_SET, 65.,-19.,0, /LAMBERT, $
898   LIMIT=[ymin,xmin+x_hlid-0.5,ymin+y_hlid,xmin+2*x_hlid-0.5], $
899   POSITION=POS, /clip , /isotropic
900
901 Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
902 contour, totprec[0:dim2d[1]-2,0:dim2d[0]-2], lon[0:dim2d[1]-2,0:dim2d[0]-2],$
903   lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, levels=r_levels, $
904   NLevels=rlevels, /Follow, /cell_fill, C_colors=Reverse(Findgen(rlevels))+1
905
906
907 colorbar, /vertical, Position = [0.915, 0.05, 0.94, 0.9], $
908 Range=[0,r_levels(rlevels-1)], NColors=rlevels, Bottom=1, /INVERTCOLORS,$
909 Color=!P.color, Divisions=rlevels/2, /right, TITLE='mm/klst'
910
911
912 Loadct, 0, /silent
913
914 oplot, strond[0,*], strond[1,*], color=0
915 ; vötn
916 a=IntArr(2)
917 a(0)=0
918 a(1)=strmid(votn_data(0),11,4)
919 FOR s=0,133 DO BEGIN
920   vatn_ll=DblArr(2,a(1))
921   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
922     vatn_ll(0,z-a(0)) = strmid(votn_data(z+1),1,10)
923     vatn_ll(1,z-a(0)) = strmid(votn_data(z+1),12,9)
924   ENDFOR
925 oplot, vatn_ll[0,*], vatn_ll[1,*], color=0
926 a(0)=a(0)+a(1)+1
927 a(1)=strmid(votn_data(a(0)),11,4)
928 ENDFOR
929 ; jöklar
930 a=IntArr(2)
931 a(0)=0
932 a(1)=strmid(joklar_data(0),11,4)
933 FOR s=0,20 DO BEGIN
934   jokl_ll=DblArr(2,a(1))
935   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
936     jokl_ll(0,z-a(0)) = strmid(joklar_data(z+1),1,10)
937     jokl_ll(1,z-a(0)) = strmid(joklar_data(z+1),12,9)
938   ENDFOR

```

```

939  oplot, jokl_ll[0,*], jokl_ll[1,*], color=0
940  a(0)=a(0)+a(1)+1
941  a(1)=strmid(joklar_data(a(0)),11,4)
942  ENDFOR
943  windbarb_eme, lon_grid[0:dim2d[1]-2,0:dim2d[0]-2],$
944    lat_grid[0:dim2d[1]-2,0:dim2d[0]-2],$
945    Hradi[0:dim2d[1]-2,0:dim2d[0]-2],$
946    Horn[0:dim2d[1]-2,0:dim2d[0]-2], $
947    LENGTH=0.02, color='black', clipnorm=pos, lim=pos
948
949  ;contour, press[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
950  ; lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, /FOLLOW, levels=prelevels
951
952  MAP_GRID, /GRID, LonDel=1, LatDel=0.5, POSISTION=pos, /BOX_AXES, lons=-26
953  ; dags + tími settur á mynd
954  strengur = strmid(dags,8,2)+'.'+strmid(dags,5,2)+'.'+strmid(dags,2,2)+ $
955  ' kl: ' + strmid(dags,11,5)+' '
956  XYOUTS, 0.05, 0.95, strengur, /Normal
957
958  ;HITI
959  Loadct, 33,Ncolors=levels, Bottom=0, /silent
960  contour, T2[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
961    lat[0:dim2d[1]-2,0:dim2d[0]-2], levels=userlevels, $
962    /overplot, /Follow, C_colors=Findgen(levels)
963
964  device,/close_file
965  abba=size(V)
966
967  ENDIF
968  ;-----09 BYRJAR-----
969  IF N_Elements(niu_9) EQ 1 OR N_Elements(allt) EQ 1 THEN BEGIN
970  ;-----
971  IF N_Elements(grisja_niu) EQ 0 THEN grisja_niu=1
972  U10_grid = congrid(U10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
973  V10_grid = congrid(V10,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
974  lon_grid = congrid(lon,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
975  lat_grid = congrid(lat,dim2d[1]/grisja_niu,dim2d[0]/grisja_niu)
976  Horn_grid = -!radeg*atan(V10_grid,U10_grid)-90
977  Hradi_grid = sqrt(V10_grid^2.+U10_grid^2.)
978  dim2d_grid=size(Hradi_grid, /dim)
979
980  IF x LT 10 THEN BEGIN
981  Device, filename='09_A0'+strtrim(string(x),2)+'ps',/color,Bits_Per_Pixel=8
982  ENDIF ELSE BEGIN
983  Device, filename='09_A'+strtrim(string(x),2)+'ps',/color,Bits_Per_Pixel=8

```

```

984 ENDELSE
985
986 MAP_SET, 65.,-19.,0, /LAMBERT, $
987   LIMIT=[ymin,xmax-x_hlid,ymin+y_hlid,xmax], $
988   POSITION=POS, /clip , /isotropic
989
990 Loadct, 21,Ncolors=rlevels, Bottom=1, /silent
991 contour, totprec[0:dim2d[1]-2,0:dim2d[0]-2], lon[0:dim2d[1]-2,0:dim2d[0]-2],$
992   lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, levels=r_levels, $
993   NLevels=rlevels, /Follow, /cell_fill, C_colors=Reverse(Findgen(rlevels))+1
994
995 colorbar, /vertical, Position = [0.915, 0.05, 0.94, 0.9], $
996 Range=[0,r_levels(rlevels-1)], NColors=rlevels, Bottom=1, /INVERTCOLORS,$
997 Color=!P.color, Divisions=rlevels/2, /right, TITLE='mm/klst'
998
999
1000 Loadct, 0, /silent
1001
1002 oplot, strond[0,*], strond[1,*], color=0
1003 ; vötn
1004 a=IntArr(2)
1005 a(0)=0
1006 a(1)=strmid(votn_data(0),11,4)
1007 FOR s=0,133 DO BEGIN
1008   vatn_ll=DblArr(2,a(1))
1009   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
1010     vatn_ll(0,z-a(0)) = strmid(votn_data(z+1),1,10)
1011     vatn_ll(1,z-a(0)) = strmid(votn_data(z+1),12,9)
1012   ENDFOR
1013   oplot, vatn_ll[0,*], vatn_ll[1,*], color=0
1014   a(0)=a(0)+a(1)+1
1015   a(1)=strmid(votn_data(a(0)),11,4)
1016   ENDFOR
1017 ; jökklar
1018 a=IntArr(2)
1019 a(0)=0
1020 a(1)=strmid(joklar_data(0),11,4)
1021 FOR s=0,20 DO BEGIN
1022   jokl_ll=DblArr(2,a(1))
1023   FOR z=a(0),a(0)+a(1)-1 DO BEGIN
1024     jokl_ll(0,z-a(0)) = strmid(joklar_data(z+1),1,10)
1025     jokl_ll(1,z-a(0)) = strmid(joklar_data(z+1),12,9)
1026   ENDFOR
1027   oplot, jokl_ll[0,*], jokl_ll[1,*], color=0
1028   a(0)=a(0)+a(1)+1

```



```

1029 a(1)=strmid(joklar_data(a(0)),11,4)
1030 ENDFOR
1031 windbarb_eme, lon_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
1032   lat_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
1033   Hradi_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
1034   Horn_grid[0:dim2d[1]-2,0:dim2d[0]-2], $
1035   LENGTH=0.02, color='black', clipnorm=pos, lim=pos
1036
1037 ;contour, press[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
1038 ; lat[0:dim2d[1]-2,0:dim2d[0]-2], /overplot, /FOLLOW, levels=prelevels
1039
1040 MAP_GRID, /GRID, LonDel=1, LatDel=0.5, POSISTION=pos, /BOX_AXES, lons=-26
1041 ; dags + tími settur á mynd
1042 strengur = strmid(dags,8,2)+'.'+strmid(dags,5,2)+'.'+strmid(dags,2,2)+ $
1043   ' kl: ' + strmid(dags,11,5)+' '
1044 XYOUTS, 0.05, 0.95, strengur, /Normal
1045
1046 ;HITI
1047 Loadct, 33,Ncolors=levels, Bottom=0, /silent
1048 contour, T2[0:dim2d[1]-2,0:dim2d[0]-2],lon[0:dim2d[1]-2,0:dim2d[0]-2], $
1049   lat[0:dim2d[1]-2,0:dim2d[0]-2], levels=userlevels, $
1050   /overplot, /Follow, C_colors=Findgen(levels)
1051
1052 device,/close_file
1053 abba=size(V)
1054
1055 ENDIF
1056
1057 ENDFOR
1058 Set_plot, thisdevice
1059 END

```

B.2 teikna_L.pro

```
1  @date2timeMM5
2  @pruneMM5
3  @periodMM5
4  @totalMM5__define
5  @colorbar
6  @bar_plot_noaxis
7  @wind_barb_ps
8
9  function shft,f
10 gg=f
11 dim = size(gg,/dim)
12 gg=rebin(gg,2*dim[0],2*dim[1])
13 gg=shift(gg,1,1)
14 gg=rebin(gg,dim[0],dim[1])
15 return,gg
16 end
17
18 PRO TEIKNA_L
19
20 ; NOKKRIR FASTAR
21 skref=6 ; lengd spáskrefs í mm5 í klst.
22 ;
23 terfile='TERRAIN/TERRAIN_DOMAIN1'
24 ; Let us begin with the mapping stuff
25 mappings,terFile,scalars=terinfo,arrays=terarr
26 ; get the lat/lon values of the dot-points
27 lat_dot = terarr.latitdot
28 lon_dot = terarr.longidot
29 ; now we calculate the i,j (floating) values of the dot-points
30 map2ij,terinfo,lat_dot,lon_dot,idot,jdot
31 ;
32 ; Time to dig into the MM5 data
33 ;
34 default,lstFile,'listi.lst'
35 b=obj_new('mm5ds',lstFile)
36 b->read
37 dim2d = b->Dim('RAIN NON')
38 dim3d = b->Dim('U')
39 b->SetIterDict,'prune' ; TÍMI
40
41 ; number of timesteps
42 nt = b->nPruned()
43 ; number of sigma levels
```

```

44 ns = dim3d[2]
45 p=b->PruneDict()
46
47 ; Let's get the topography data
48 ter_tmp = b->data(0,'TERRAIN',/array)
49 iinf     = where(finite(ter_tmp, /NaN),cinf)
50 if cinf gt 0 then ter_tmp[iinf]=-0.001
51 ter      = transpose(ter_tmp)
52
53 ; náum í lista yfir þá staði sem teikna skal spárit fyrir
54 stadir_file = '/home/sumar1/idl/kort/vedurstod.lst'
55 linufjoldi=File_Lines(stadir_file)
56 stadir_tmp=StrArr(linufjoldi)
57 OpenR, lun, stadir_file, /Get_Lun
58 ReadF, lun, stadir_tmp
59 Free_Lun, lun
60
61 ll_stadir = FltArr(3,linufjoldi)
62 nofn      = StrArr(2,linufjoldi)
63 for j=0,linufjoldi-1 DO BEGIN
64 IF strmid(stadir_tmp(j), 0,1) EQ 1 THEN BEGIN
65 ll_stadir(0,j) = strmid(stadir_tmp(j), 2,5)
66 ll_stadir(1,j) = strmid(stadir_tmp(j), 8,5)
67 ll_stadir(2,j) = strmid(stadir_tmp(j),14,1)
68 nofn(0,j)      = strmid(stadir_tmp(j),16,10)
69 nofn(1,j)      = strmid(stadir_tmp(j),26,40)
70 ENDIF
71 IF strmid(stadir_tmp(j), 0,1) EQ 2 THEN BEGIN
72 ll_stadir(0,j) = strmid(stadir_tmp(j), 2,2)+$
73                strmid(stadir_tmp(j), 5,2)/60.
74 ll_stadir(1,j) = strmid(stadir_tmp(j), 8,2)+$
75                strmid(stadir_tmp(j),11,2)/60.
76 ll_stadir(2,j) = strmid(stadir_tmp(j),14,1)
77 nofn(0,j)      = strmid(stadir_tmp(j),16,10)
78 nofn(1,j)      = strmid(stadir_tmp(j),25,40)
79 ENDIF
80 IF strmid(stadir_tmp(j), 0,1) EQ 3 THEN BEGIN
81 ll_stadir(0,j) = strmid(stadir_tmp(j), 2,2)+$
82                strmid(stadir_tmp(j), 5,2)/60+ $
83                strmid(stadir_tmp(j), 8,2)/3600
84 ll_stadir(1,j) = strmid(stadir_tmp(j),11,2)+$
85                strmid(stadir_tmp(j),14,2)/60+ $
86                strmid(stadir_tmp(j),17,2)/3600
87 ll_stadir(2,j) = strmid(stadir_tmp(j),20,1)
88 nofn(0,j)      = strmid(stadir_tmp(j),22,7)

```

```

89  nofn(1,j)      = strmid(stadir_tmp(j),30,40)
90  ENDIF
91  IF strmid(stadir_tmp(j), 0,1) EQ 4 THEN BEGIN
92  ll_stadir(0,j) = strmid(stadir_tmp(j), 2,2)+strmid(stadir_tmp(j), 5,6)/60
93  ll_stadir(1,j) = strmid(stadir_tmp(j),12,2)+strmid(stadir_tmp(j),15,6)/60
94  ll_stadir(2,j) = strmid(stadir_tmp(j),22,1)
95  nofn(0,j)      = strmid(stadir_tmp(j),24,8)
96  nofn(1,j)      = strmid(stadir_tmp(j),33,40)
97  ENDIF
98  IF strmid(stadir_tmp(j), 0,1) EQ 5 THEN BEGIN
99  ll_stadir(0,j) = 0
100 ll_stadir(1,j) = 0
101 ll_stadir(2,j) = 0
102 nofn(0,j)      = 0
103 nofn(1,j)      = strmid(stadir_tmp(j),2,40)
104 ENDIF
105
106 ENDFOR
107
108 ;map2ij,terinfo,ll_stadir(0,*),ll_stadir(1,*),is,js
109 dags=StrArr(nt)
110 dagur=fltArr(nt)
111 data=Db1Arr(8,nt,linufjoldi)
112
113
114 FOR x=0,nt-2 DO BEGIN
115   lon=transpose(lon_dot)
116   lat=transpose(lat_dot)
117   u10_tmp=b->data(p.keys[x+1], 'U10', /array)
118   v10_tmp=b->data(p.keys[x+1], 'V10', /array)
119   u10=transpose(shft(U10_tmp))
120   v10=transpose(shft(v10_tmp))
121   t2_tmp=b->data(p.keys[x+1], 'T2', /array)
122   T2_kelvin=transpose(t2_tmp)
123   T2=T2_kelvin-273.15
124   clw_tmp=b->data(p.keys[x+1], 'clw', /array)
125   clw=transpose(clw_tmp)
126   fnon0=b->data(p.keys[x], 'RAIN NON', /array)
127   fnon1=b->data(p.keys[x+1], 'RAIN NON', /array)
128   fcon0=b->data(p.keys[x], 'RAIN CON', /array)
129   fcon1=b->data(p.keys[x+1], 'RAIN CON', /array)
130   fcon=fcon1-fcon0
131   fnon=fnon1-fnon0
132 ; Some cleaning up and unit conversion
133   totprec=10.0*(fnon+fcon) ;change into mm

```

```

134 ; Transpose all the fields so as it fits our way of thinking:-)
135   totprec=transpose(totprec)
136   t_tmp=b->data(p.keys[x+1], 'T', /array)
137   T_kelvin=transpose(t_tmp)
138   T=T_kelvin-273.15
139 ; clw (cloud water) summað upp um öll ns
140 ;SH = fltarr(dim2d)
141   SH = total(clw, 1)
142
143 ;
144 ; Prýstingspælingar!
145 ;
146   ptop    = 10000
147   PP_tmp  = b->data(p.keys[x+1], 'PP', /array)
148   PP      = transpose(PP_tmp)
149   PSTAR_tmp = b->data(p.keys[x+1], 'PSTARCRS', /array)
150   PSTAR    = transpose(PSTAR_tmp)
151   sigma_tmp = b->data(p.keys[x+1], 'SIGMAH', /array)
152   sigma     = sigma_tmp
153   Q_tmp     = b->data(p.keys[x+1], 'Q', /array)
154   Q        = transpose(Q_tmp)
155   press=reduce_to_slp((sigma(ns-1)*pstar+pp[ns-1,*,*]+ptop)/100, $
156                       T_kelvin[ns-1,*,*], Q[ns-1,*,*], ter)
157   e1 = Alog((q/0.62197)*(press/100.)/(1.+(q/0.62197)))
158   td = (243.5*e1-440.8)/(19.48-e1)
159
160 ; tímastrengur á mynd
161 ; dags=strmid(p.keys[x+1],0,13)
162
163   Horn = -!radeg*atan(V10,U10)-90
164   Hradi = sqrt(V10^2.+U10^2.)
165 ; dags + tími
166   dags(x) = p.keys[x+1]
167   dagur(x)= julday(strmid(dags(x),5,2),strmid(dags(x),8,2),$
168                   strmid(dags(x),0,4),strmid(dags(x),11,2),0,0)
169
170
171 FOR k=0,linufjoldi-1 DO BEGIN
172   imn=floor(min(idot[k]))
173   imx=ceil(max(idot[k]))
174   jmn=floor(min(jdot[k]))
175   jmx=ceil(max(jdot[k]))
176   st=size(idot, /dim)
177
178   nx1 = imx-imn+1

```

```

179     ny1 = jmx-jmn+1
180
181 ; þurfum að velja ákveðna punkta úr safninu...
182
183 ; Now we have to find the point that is closest to our original
184 ; Lat/Lon value
185 ; We do this by changing the original Lat/Lon vector into a
186 ; complex array, do the same for our new rebin-ed points
187     latlon_in=complex(ll_stadir[0,k],-ll_stadir[1,k]) ; ath for lykkju
188     latlon_cross=complex(lat,lon)
189 ; Now we find the rebin-ed point that is closest to our latlon_in value
190     minval=MIN(ABS(latlon_cross-latlon_in))
191     point=WHERE(ABS(latlon_cross-latlon_in) EQ minval)
192
193 ;print, point,ll_stadir[0,k],ll_stadir[1,k];,lat,lon
194 ; Finally get the x,y position of "point"
195     pointx=point mod st(1)
196     pointy=Fix(point/st(1))
197
198 ;     pointx=point mod nx1
199 ;     pointy=Fix(point/nx1)
200 ;print, st
201 ;print, pointx,pointy
202 ; Take out the value from pointx,pointy
203     pntprec=totprec [pointx,pointy]
204     pntt2=t2[pointx,pointy]
205     pnthradi=hradi[pointx,pointy]
206     pnthorn=horn[pointx,pointy]
207     pntlat=lat[pointx,pointy]
208     pntlon=lon[pointx,pointy]
209     pnttd=td[pointx,pointy]
210
211 ;     pntPress=(sigma[s]*pntPstar+pntPP+ptop)
212 ;     pntQ=Q[s,skewT_hnit[0,k],skewT_hnit[1,k]]
213 ;     pnttd=td(pntQ,pntpress)
214
215     pntSH=SH[pointx,pointy]
216 ; þarf að raða inn í fylki
217 ;
218 ;     dags(x)=strmid(p.keys[x+1],0,13)
219 ;     dagur(x)=julday(strmid(dags(x),5,2),strmid(dags(x),8,2),$
220 ;                     strmid(dags(x),0,4),strmid(dags(x),11,2),0,0)
221     data(0,x,k)=pntlat
222     data(1,x,k)=pntlon
223     data(2,x,k)=pntprec

```

```

224     data(3,x,k)=pntt2
225     data(4,x,k)=pnthradi
226     data(5,x,k)=pnthorn
227     data(6,x,k)=pnttd
228 ;     data(7,x,k)=pntSH
229
230 ENDFOR
231 ENDFOR
232
233 ; litur fyrir bar_plot
234 litur=findgen(ns)*0+30
235 dagurr=findgen(nt-1)*skref ; á að vera nt en ekki 24
236 ;print, dagur, dagurr
237 ;print, data(3,0,7)
238
239 thisdevice = !D.Name
240 Set_Plot, 'PS'
241
242 FOR z=0,linufjoldi-1 DO BEGIN
243
244 max_t=max(data(3,*,Z))
245 maxy = ceil(max_t)+(5-ceil(max_t) mod 5)
246 bil=[maxy-20,maxy]
247
248 max_p=max(data(2,*,z))
249 IF max_p GT 4 THEN bilyy=[0,ceil(max_p)+(5-ceil(max_p)mod 5)] ELSE bilyy=[0,4]
250
251 IF z LT 10 THEN BEGIN
252 Device, filename='sparit0' + strtrim(string(z),2) + '.ps', /color,Bits=8
253 ENDIF ELSE BEGIN
254 Device, filename='sparit' + strtrim(string(z),2) + '.ps', /color,Bits=8
255 ENDELSE
256
257 loadct, 12, /silent
258
259 ; "hiti plottaður"
260 plot, dagurr, data(3,*,Z), YTickLen=1, Ygridstyle=2,$
261     Ytitle='Hitastig [°C]', YCharSize=1.1, XCharSize=1.1, $
262     Position = [0.1, 0.3, 0.9, 0.95], Xticks=6, Yrange=bil, YStyle=9, $
263     background=0, XRange=[0,24], /nodata, $
264     XStyle=1, xminor=3, xtickinterval=3;, XTickV=[3,6,9,12,15,18,21]
265
266 ; Skýjahula
267 ;print, 'meðaltal: ',total(sh)/n_elements(sh)/40 , 'max: ', max(sh)
268 ;print, 'meðaltal_clw: ',total(clw)/n_elements(clw) , ' max: ', max(clw)

```

```

269 loadct, 0 , /silent
270
271 ;Axis, Yaxis=1, /save, Yrange=[0,1], YStyle=4
272 ;X_birt=[!X.Range[0],dagur[0:23],!X.Range[1],!X.Range[1], !X.Range[0]]
273 ;; hér á eftir á 24 auðvitað að vera nt
274 ;Y_birt=[data(7,0,Z),reform(data(7,0:23,Z)),data(7,23,z),!Y.Range[1], !Y.Range[1]]
275 ;polyfill, X_birt ,Y_birt, color=128
276
277
278 loadct, 12, /silent
279 ; vindörvar
280 X_normal = 0.1+0.8/(nt*2) ; 24 eiga að vera nt
281 Y_normal = 0.15
282 FOR j=0,nt-1 DO BEGIN ; á að vera nt-1
283 wind_barb_ps, data(4,j,z), data(5,j,z), X_normal, Y_normal, /NoErase, $
284         Size=0.05, thick=1.5, /Normal, /ms ;, /nocircle
285 X_normal = X_normal + 0.8/nt ; 24 á að vera nt
286 print, j
287 ENDFOR
288
289 ;regn
290 Axis, Yaxis=1, YTitle='Úrkoma [mm/klst]', /save, Yrange=bilyy, YCharSize=1.1
291 bar_plot_noaxis, data(2,0:*,z),/overplot, colors=litur, /noaxis; hér á að vera * en e
292
293 ; dagseting og rammi
294 strengur=nofn(1,z)+' '+strmid(dags(0),8,2)+'.'+strmid(dags(0),5,2)+'.'+strmid(dags(0)
295
296 XYOUTS, 0.05, 0.96, strengur, /Normal, charsize=1.3
297 Plots, [0,1,1,0,0], [0,0,1,1,0], /normal, thick=3
298
299 ; hitagildin sett inn í öðrum lit
300 Axis, Yaxis=1, /save, Yrange=bil, YStyle=4
301 oplot, dagurr, data(3,*,z), color=200, zvalue=0
302 oplot, dagurr, data(6,*,z), color=100, zvalue=0
303
304 device,/close_file
305
306
307 ENDFOR
308 ;print, data(6,*,0) ; útprentun á TD ??
309 Set_plot, thisdevice
310 END

```


B.3 teikna_skewT.pro

```
1  @date2timeMM5
2  @pruneMM5
3  @periodMM5
4  @totalMM5__define
5  @colorbar
6  @skew-t
7  @plot_skewt
8  @td
9  @wind_barb_ps
10
11 function shft,f
12 gg=f
13 dim = size(gg,/dim)
14 gg=rebin(gg,2*dim[0],2*dim[1])
15 gg=shift(gg,1,1)
16 gg=rebin(gg,dim[0],dim[1])
17 return,gg
18 end
19
20 function shft3d,f
21 gg=f
22 dim = size(gg,/dim)
23 gg=rebin(gg,2*dim[0],2*dim[1],dim[2])
24 gg=shift(gg,1,1,0)
25 gg=rebin(gg,dim[0],dim[1],dim[2])
26 return,gg
27 end
28
29 PRO TEIKNA_skewT, KEF=kef, $ ; plotta bara fyrir Keflavík
30         NET=net      ; stærð nets, sjálfgefið er [5,7], þ.e. 5*7
31
32 ; Hnattstaða keflavíkurstöðvarinnar
33 kef = [63.97,-22.59]
34 IF N_Elements(NET) EQ 0 THEN net = [5,7]
35
36 ;
37 terfile='/mnt/haugarok/home/rassmus/mm5suite/v3-6/TERRAIN/TERRAIN_DOMAIN1_Kaisa'
38 ; Let us begin with the mapping stuff
39 mappings,terFile,scalars=terinfo,arrays=terarr
40 ; get the lat/lon values of the dot-points
41 lat_dot = terarr.latitdot
42 lon_dot = terarr.longidot
43 ; now we calculate the i,j (floating) values of the dot-points
```

```

44 map2ij,terinfo,lat_dot,lon_dot,idot,jdot
45 ;
46 ; Time to dig into the MM5 data
47 ;
48 default,lstFile,'listi.lst'
49 b=obj_new('mm5ds',lstFile)
50 b->read
51 dim2d = b->Dim('RAIN NON')
52 dim3d = b->Dim('U')
53 b->SetIterDict,'prune' ; TÍMI
54
55 ; number of timesteps
56 nt = b->nPruned()
57 ; number of sigma levels
58 ns = dim3d[2]
59 p=b->PruneDict()
60
61 ; Let's get the topography data
62 ter_tmp = b->data(0,'TERRAIN',/array)
63 iinf = where(finite(ter_tmp, /NaN),cinf)
64 if cinf gt 0 then ter_tmp[iinf]=-0.001
65 ter = transpose(ter_tmp)
66
67 ;map2ij,terinfo,ll_stadir(0,*),ll_stadir(1,*),is,js
68 ;linufjoldi=100
69 dags=StrArr(nt)
70 dagur=fltArr(nt)
71 data=Db1Arr(7,nt,ns,net[0]*net[1])
72
73 FOR x=0,nt-2 DO BEGIN
74 lon=transpose(lon_dot)
75 lat=transpose(lat_dot)
76 ; Transpose all the fields so as it fits our way of thinking:-)
77 t_tmp=b->data(p.keys[x+1],'T',/array)
78 T_kelvin=transpose(t_tmp)
79 T=T_kelvin-273.15
80 u_tmp=b->data(p.keys[x+1],'U',/array)
81 v_tmp=b->data(p.keys[x+1],'V',/array)
82 u=transpose(shft3d(U_tmp))
83 v=transpose(shft3d(v_tmp))
84 ;
85 ; Þrýstingsþælingar!
86 ;
87 ptop = 10000 ; veit þó ekki fyrir víst!
88 PP_tmp = b->data(p.keys[x+1],'PP',/array)

```

```

89     PP      = transpose(PP_tmp)
90     PSTAR_tmp = b->data(p.keys[x+1], 'PSTARCRS', /array)
91     PSTAR     = transpose(PSTAR_tmp)
92     sigma_tmp = b->data(p.keys[x+1], 'SIGMAH', /array)
93     sigma     = sigma_tmp
94     Q_tmp     = b->data(p.keys[x+1], 'Q', /array)
95     Q        = transpose(Q_tmp)
96     ; press=reduce_to_slp((sigma(ns-1)*pstar+pp[ns-1,*,*]+ptop)/100, $
97     ;                      T_kelvin[ns-1,*,*], Q[ns-1,*,*], ter)
98     ; PRESS=(sigma*pstar+pp+ptop)/100
99     ; e1 = Alog((q/0.62197)*(press/100.)/(1.+(q/0.62197)))
100    ; td = (243.5*e1-440.8)/(19.48-e1)
101
102    ; tímastrengur á mynd
103    ; dags=strmid(p.keys[x+1],0,13)
104
105    Horn = -!radeg*atan(V,U)-90
106    Hradi = sqrt(V^2.+U^2.)
107    ; dags + tími
108    dags(x) = p.keys[x+1]
109    dagur(x)= julday(strmid(dags(x),5,2),strmid(dags(x),8,2),$
110                   strmid(dags(x),0,4),strmid(dags(x),11,2),0,0)
111
112
113    ; We do this by changing the original Lat/Lon vector into a
114    ; complex array, do the same for our new rebin-ed points
115    latlon_in=complex(kef[0],kef[1]) ; ath for lykkju
116    latlon_cross=complex(lat,lon)
117    ; Now we find the rebin-ed point that is closest to our latlon_in value
118    minval=MIN(ABS(latlon_cross-latlon_in))
119    point=WHERE(ABS(latlon_cross-latlon_in) EQ minval)
120    ; Finally get the x,y position of "kef"
121    pointx=point mod dim2d(1)
122    pointy=Fix(point/dim2d(1))
123    ; finnum þá punkta sem teikna á út frá keflavíkurpunkti 5X8 net
124    skewT_hnit=IntArr(2,net[0]*net[1])
125    FOR j=0,net[0]-1 DO BEGIN
126    FOR k=0,net[1]-1 DO BEGIN
127    IF ((dim2d[0]/net[0])*(j-1)+(pointy)) LT 58 AND $
128    ((dim2d[0]/net[0])*(j-1)+(pointy)) GE 0 THEN BEGIN
129    IF ((dim2d[1]/net[1])*(k-1)+(pointx)) LT 79 AND $
130    ((dim2d[1]/net[1])*(k-1)+(pointx)) GE 0 THEN BEGIN
131    skewT_hnit[1,k+j*net[1]]=[(dim2d[0]/net[0])*(j-1)+(pointy)]
132    skewT_hnit[0,k+j*net[1]]=[(dim2d[1]/net[1])*(k-1)+(pointx)]
133    ENDIF

```

```

134  ENDIF
135  ENDFOR
136  ENDFOR
137
138  ; sækjum þá gögnin fyrir ofangreinda punkta!
139  FOR k=0,(net[0]*net[1])-1 DO BEGIN
140      pntlat=lat[skewT_hnit[0,k],skewT_hnit[1,k]]
141      pntlon=lon[skewT_hnit[0,k],skewT_hnit[1,k]]
142      pntPstar=Pstar[skewT_hnit[0,k],skewT_hnit[1,k]]
143  FOR s=0,ns-1 DO BEGIN
144      pnthradi=hradi[s,skewT_hnit[0,k],skewT_hnit[1,k]]
145      pnthorn=horn[s,skewT_hnit[0,k],skewT_hnit[1,k]]
146      pntPP=PP[s,skewT_hnit[0,k],skewT_hnit[1,k]]
147      pntQ=Q[s,skewT_hnit[0,k],skewT_hnit[1,k]]
148      pntPress=(sigma[s]*pntPstar+pntPP+ptop)
149      pntT=T[s,skewT_hnit[0,k],skewT_hnit[1,k]]
150      pntQ=Q[s,skewT_hnit[0,k],skewT_hnit[1,k]]
151      ; e1 = Alog10((pntQ/0.62197)*(pntPress/100.)/(1.+(pntQ/0.62197)))
152      ; pnttd=(243.5*e1-440.8)/(19.48-e1)+273.15
153      pnttd=td(pntQ,pntpress)
154
155  ; röðum inn í fylki
156      data(0,x,s,k)=pntlat
157      data(1,x,s,k)=pntlon
158      data(2,x,s,k)=pnthradi
159      data(3,x,s,k)=pnthorn
160      data(4,x,s,k)=pntPress
161      data(5,x,s,k)=pntT
162      data(6,x,s,k)=pnttd
163  ENDFOR
164  ENDFOR
165  ENDFOR
166
167
168  P_w=[100,200,300,400,500,600,700,800,900,1000]
169
170  FOR x=0,nt-2 DO BEGIN
171  FOR z=0,(net[0]*net[1])-1 DO BEGIN
172
173  w_sp=interpol(reform(data(2,x*,z)),reform(data(4,x*,z))/100.,p_w)
174  w_dir=interpol(reform(data(3,x*,z)),reform(data(4,x*,z))/100.,p_w)
175  Set_Plot, 'PS'
176
177  t_end=reform(data(5,x*,z))
178  td_end=reform(data(6,x*,z))

```

```

179 p_l=reform(data(4,x*,z))/100.
180
181 ; dagseting og rammi
182 IF z NE 8 THEN BEGIN
183 strengur=strmid(dags(x),8,2)+'.'+strmid(dags(x),5,2)+'.'+strmid(dags(x),2,2)+' kl: '
184 ENDIF ELSE BEGIN
185 strengur='Keflavík' + ' ' + strmid(dags(x),8,2)+'.'+strmid(dags(x),5,2)+'.'+strmid(da
186 ENDELSE
187
188
189 skewt,[-40, 40], everyT=10, everyDA=10, everySA=2, everyW=2, title=strengur
190
191 ;help, T_end, td_end, p_l, w_dir, w_sp, P_w
192 plot_skewt, T_end, td_end, p_l, w_dir, w_sp, P_w
193
194
195 IF x LT 10 THEN BEGIN
196 IF z LT 10 THEN BEGIN
197 spawn,'mv skewt.ps skewT0' + strtrim(string(z),2) + '_0' + $
198     strtrim(string(x),2) + '.ps'
199 ENDIF ELSE BEGIN
200 spawn,'mv skewt.ps skewT' + strtrim(string(z),2) + '_0' + $
201     strtrim(string(x),2) + '.ps'
202 ENDELSE
203 ENDIF ELSE BEGIN
204 IF z LT 10 THEN BEGIN
205 spawn,'mv skewt.ps skewT0' + strtrim(string(z),2) + '_' + $
206     strtrim(string(x),2) + '.ps'
207 ENDIF ELSE BEGIN
208 spawn,'mv skewt.ps skewT' + strtrim(string(z),2) + '_' + $
209     strtrim(string(x),2) + '.ps'
210 ENDELSE
211 ENDELSE
212
213 ENDFOR
214 print, x, '/', nt-2
215 ENDFOR ; x-for
216
217 ;print, data(6,*,0) ; útprentun á TD ??
218
219 END

```