In [ ]:
```
1
```

Effect of TMAR filter (RH100)

In [1]:
```
1  # Load modules
2  using Plots
3  using NCDatasets
4  using Statistics: mean
5  using DelimitedFiles
6  using PrettyTables
7  using PaddedViews
8  using Dates
9  using Printf
10
```

In [2]:
```
1
2  # Useful functions
3  function get_var(file_name, var_name, t_spinup, nan_fill_value);
4      ds = NCDataset(file_name, "r");
5      var = ds[var_name][:];
6      data=var[:, :, :, t_spinup:end]; # cut out the spinup;
7      replace!(data, NaN=>nan_fill_value)
8      return data
9      none
10 end
11
12 function get_zonal_mean(file_name, var_name, nan_fill_value);
13     ds = NCDataset(file_name, "r");
14     var = ds[var_name][:];
15     data_mean = mean( var[:,:,:,:], dims=1); # lon, lat,lev, time
16     replace!(data_mean, NaN=>nan_fill_value)
17     return data_mean[1,:,:,:]
18     none
19 end
20
21 function get_slice(
22                     file_name, var_name, nan_fill_value,
23                     lon_i1, lon_12, lat_i1, lat_i2, lev_i1, lev_i2, t_i1, t_i2
24
25     ds = NCDataset(file_name, "r");
26     var = ds[var_name][:];
27     data = (var[lon_i1:lon_12, lat_i1:lat_i2, lev_i1:lev_i2, t_i1:t_i2]); # c
28     replace!(data, NaN=>nan_fill_value)
29     return data
30     none
31 end
32
33 function get_coords(file_name)
34     ds = NCDataset(file_name, "r");
35     lon = ds["long"][:];
36     lat = ds["lat"][:];
37     lev = ds["level"][:] / 1e3; # height in kilometers
38     time = ds["time"][:];  # time
39     close(ds)
40     return lon, lat, lev, time
41     none
42 end
43
44 function get_min_max(var);
45     vmax = maximum(filter(!isnan,var))
46     vmin = minimum(filter(!isnan,var))
47     return vmin,vmax
48     none
49 end
50
```

```
51 function get_short_expname(fname, var_code_1)
52     driver = fname
53     exp_name =  split(split(driver,var_code_1)[2],"_")[1]
54     return exp_name
55 end
56
```

Out[2]:
```
get_short_expname (generic function with 1 method)
```

In [16]:
```
1 fnames
```

Out[16]:
```
2-element Array{String,1}:
 "ctrl_hier_RH100_q_active_np128_relax60_diffn_none_remove_q_none_AtmosGCMDefaul
t_2020-08-24T15.19.01.998.nc"
 "hier_RH100_q_active_np128_relax60_noTMAR_diffn_none_remove_q_none_AtmosGCMDefa
ult_2020-08-24T16.29.05.754.nc"
```

In [24]:
```
1  # Specify needed directories and filenames: these will be replaced automatica
2  CLIMA_ANALYSIS = "/central/scratch/elencz/output/hier_RH100_q_active_np128_re
3  CLIMA_NETCDF = "/central/scratch/elencz/output/SA_crash_data/100RHtmar"
4  CLIMA_LOGFILE = "/central/scratch/elencz/output/hier_RH100_q_active_np128_rel
5
6  # Get the current and previous  GCM netcdf file names in the CLIMA_NETCDF dir
7  fnames = filter(x -> occursin(".nc", x), readdir( CLIMA_NETCDF ) );
8
9
10 # set file name
11 filename = "$CLIMA_NETCDF/"fnames[1]
12
13 # print data info:
14 ds = NCDataset(filename, "r")
15
```

Out[24]:
```
NCDataset: /central/scratch/elencz/output/SA_crash_data/100RHtmar/ctrl_hier_RH10
0_q_active_np128_relax60_diffn_none_remove_q_none_AtmosGCMDefault_2020-08-24T15.
19.01.998.nc
Group: /

Dimensions
   long = 361
   lat = 181
   level = 31
   time = 18

Variables
  long   (361)
    Datatype:    Float64
    Dimensions:  long
    Attributes:
     units                = degrees_east
     long_name            = longitude

  lat    (181)
```

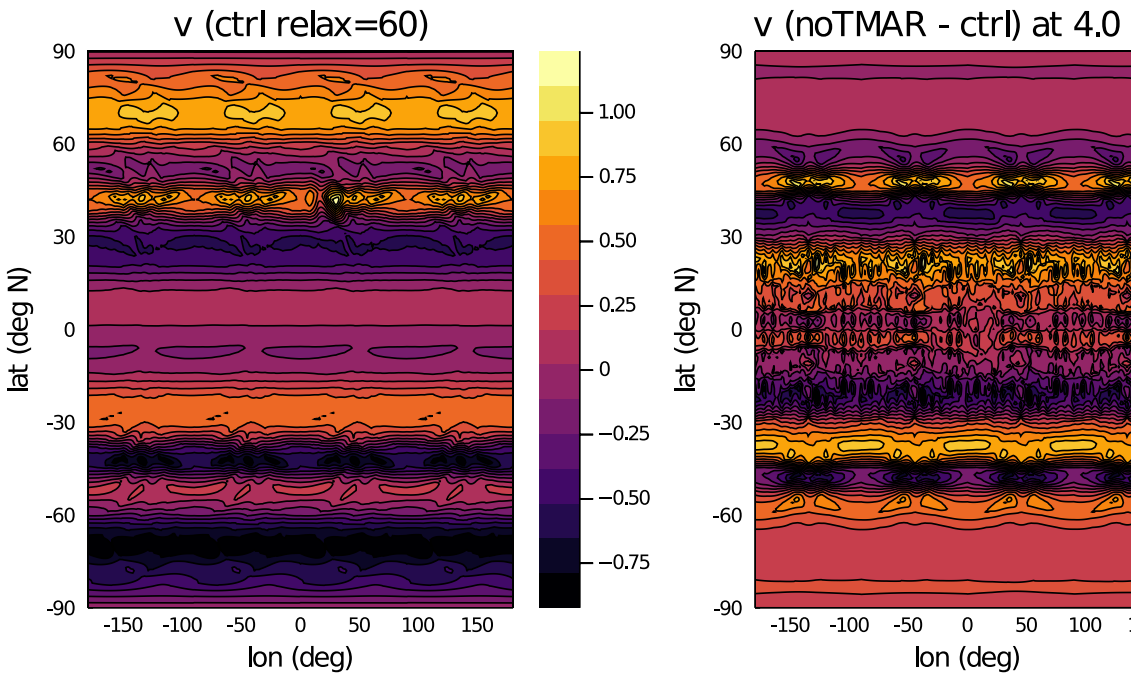
In [30]:
```
1  # get coordinates
2  lon, lat, lev, time = get_coords(filename);
3
4  nan_fill_value = -9999
5  nexp = size(fnames);
6
7
8
9
10
11
```

```
UndefVarError: var code 2 not defined
```

In [ ]:
```
1
2
```

Vertical Slices

In [31]:
```
1  #Slices
2  t_slice = 3
3  z_slice = 5
4  var_name = "v"
5
6  # vertical
7  lon_i1, lon_12, lat_i1, lat_i2, lev_i1, lev_i2, t_i1, t_i2 = [1, length(lon),
8
9
10
11 # control
12 hs_c = get_slice( filename, var_name, nan_fill_value, lon_i1, lon_12, lat_i1,
13
14 p_ctrl = contourf( lon, lat, (hs_c[:,:,1,1])', title="$var_name (ctrl relax=6
15 #
16 # Make anomaly plots and save them in an array
17 plot_array = Any[p_ctrl]; # can type this more strictly
18 for i in 2:nexp[1]
19   hs = get_slice( "$CLIMA_NETCDF/"fnames[i], var_name, nan_fill_value, lon_i1
20   exp_name = get_short_expname(fnames[i], "relax60_")
21   z_in_km = lev[z_slice]
22   one_plot = contourf( lon, lat, ( (hs[:,:,1,1]-hs_c[:,:,1,1])*1.0 )', title=
23   push!(plot_array,one_plot); # make a plot and add it to the plot_array
24 end
25
26 fig=plot(plot_array... , layout=(1, 2), size=(800, 400) )
27 #savefig(fig, string("$CLIMA_ANALYSIS/plot_$var_name","_hovmoller_sens.pdf"))
28
29
30
31 display(fig)
32
33
34
35
```



In [27]:
```
mean( (hs[:,:,1,1]-hs_c[:,:,1,1])*1.0 )
```

Out[27]: -0.00020008007930608969

Zonal means

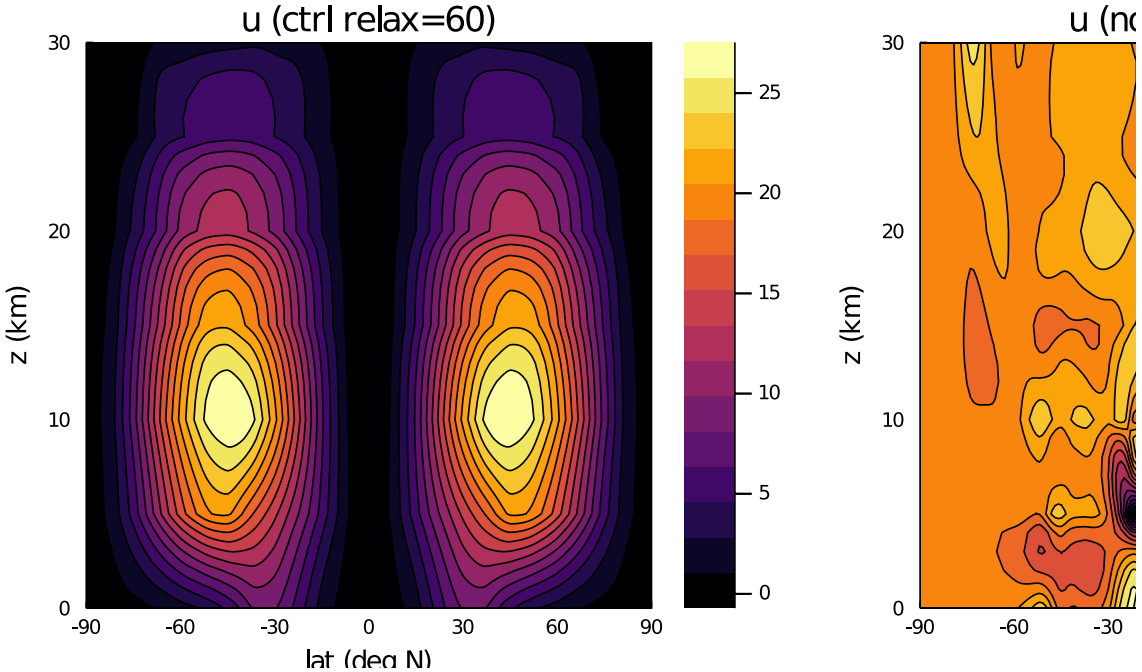
In [32]:
```
# get_zonal_means
t_slice = 3
var_name = "u"

zm_c = get_zonal_mean(filename, var_name, nan_fill_value)
p_ctrl = contourf( lat, lev, (zm_c[:,:,t_slice])' , title="$var_name (ctrl re

# Make anomaly plots and save them in an array
plot_array = Any[p_ctrl]; # can type this more strictly
for i in 2:nexp[1]
  zm = get_zonal_mean( "$CLIMA_NETCDF/"fnames[i], var_name, nan_fill_value);
  exp_name = get_short_expname(fnames[i], "relax60_")
  one_plot = contourf( lat, lev, (zm[:,:,t_slice]-zm_c[:,:,t_slice])', title=
  push!(plot_array,one_plot); # make a plot and add it to the plot_array
end

fig=plot(plot_array... , layout=(1, 2), size=(1000, 400) )
#savefig(fig, string("$CLIMA_ANALYSIS/plot_$var_name","_hovmoller_sens.pdf"))
display(fig)

```

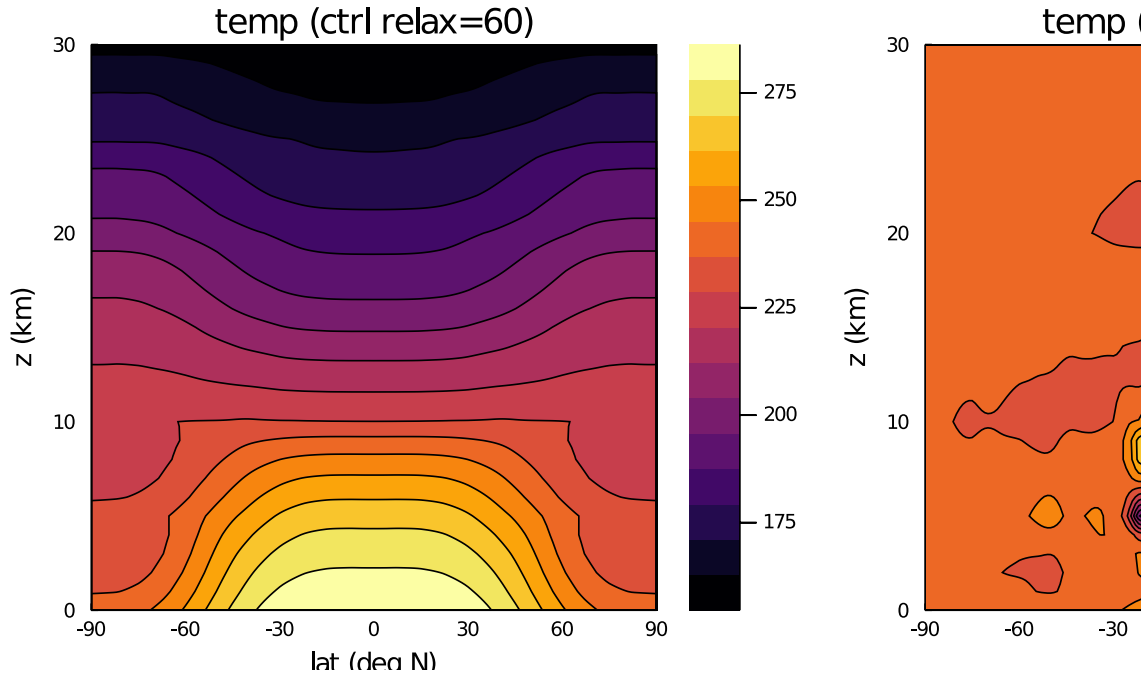


In [37]:
```
# get_zonal_mean: T
t_slice = 3
var_name = "temp"

zm_c = get_zonal_mean(filename, var_name, nan_fill_value)
p_ctrl = contourf( lat, lev, (zm_c[:,:,t_slice])' , title="$var_name (ctrl re

# Make anomaly plots and save them in an array
plot_array = Any[p_ctrl]; # can type this more strictly
for i in 2:nexp[1]
  zm = get_zonal_mean( "$CLIMA_NETCDF/"fnames[i], var_name, nan_fill_value);
  exp_name = get_short_expname(fnames[i], "relax60_")
  one_plot = contourf( lat, lev, (zm[:,:,t_slice]-zm_c[:,:,t_slice])', title=
  push!(plot_array,one_plot); # make a plot and add it to the plot_array
end
```

```
16
17 fig=plot(plot_array... , layout=(1, 2), size=(1000, 400) )
18 #savefig(fig, string("$CLIMA_ANALYSIS/plot_$var_name","_hovmoller_sens.pdf"))
19 display(fig)
20
21
```



In [53]:
```
1
```

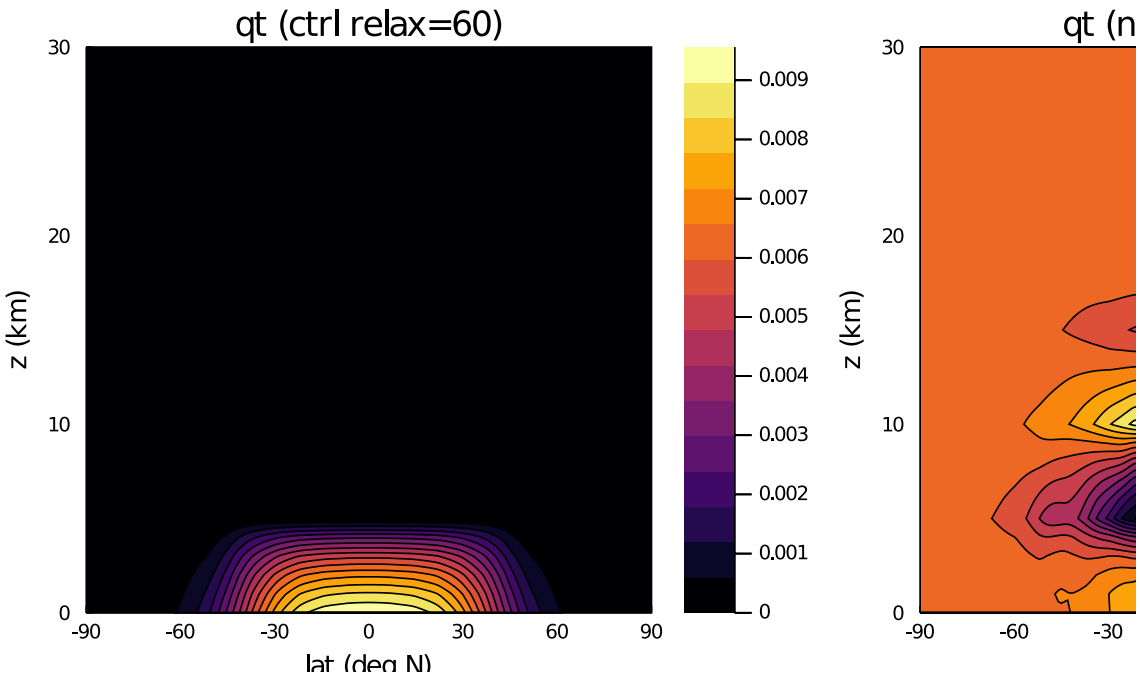
In [34]:

```
# get_zonal_mean: qt
t_slice = 3
var_name = "qt"

zm_c = get_zonal_mean(filename, var_name, nan_fill_value)
p_ctrl = contourf( lat, lev, (zm_c[:,:,t_slice])' , title="$var_name (ctrl re

# Make anomaly plots and save them in an array
plot_array = Any[p_ctrl]; # can type this more strictly
for i in 2:nexp[1]
  zm = get_zonal_mean( "$CLIMA_NETCDF/"fnames[i], var_name, nan_fill_value);
  exp_name = get_short_expname(fnames[i], "relax60_")
  one_plot = contourf( lat, lev, (zm[:,:,t_slice]-zm_c[:,:,t_slice])', title=
  push!(plot_array,one_plot); # make a plot and add it to the plot_array
end

fig=plot(plot_array... , layout=(1, 2), size=(1000, 400) )
#savefig(fig, string("$CLIMA_ANALYSIS/plot_$var_name","_hovmoller_sens.pdf"))
display(fig)
```



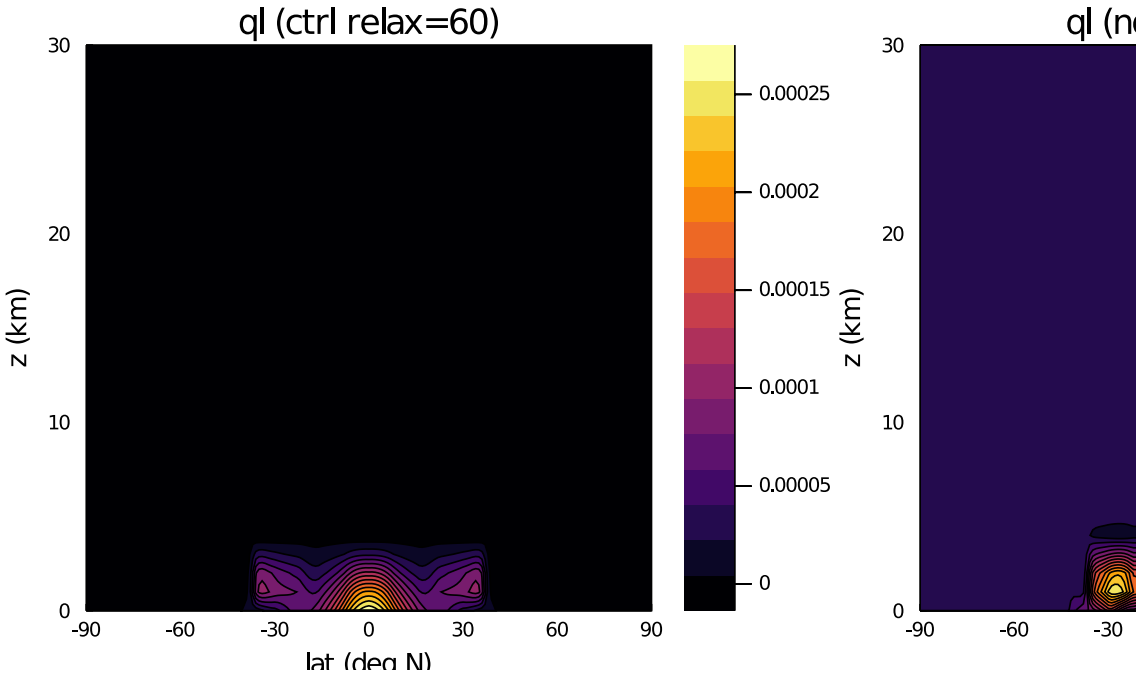
In [35]:

```
# get_zonal_mean: ql
t_slice = 3
var_name = "ql"

zm_c = get_zonal_mean(filename, var_name, nan_fill_value)
p_ctrl = contourf( lat, lev, (zm_c[:,:,t_slice])' , title="$var_name (ctrl re

# Make anomaly plots and save them in an array
plot_array = Any[p_ctrl]; # can type this more strictly
for i in 2:nexp[1]
  zm = get_zonal_mean( "$CLIMA_NETCDF/"fnames[i], var_name, nan_fill_value);
  exp_name = get_short_expname(fnames[i], "relax60_")
  one_plot = contourf( lat, lev, (zm[:,:,t_slice]-zm_c[:,:,t_slice])', title=
  push!(plot_array,one_plot); # make a plot and add it to the plot_array
end

fig=plot(plot_array... , layout=(1, 2), size=(1000, 400) )
#savefig(fig, string("$CLIMA_ANALYSIS/plot_$var_name","_hovmoller_sens.pdf"))
display(fig)
```



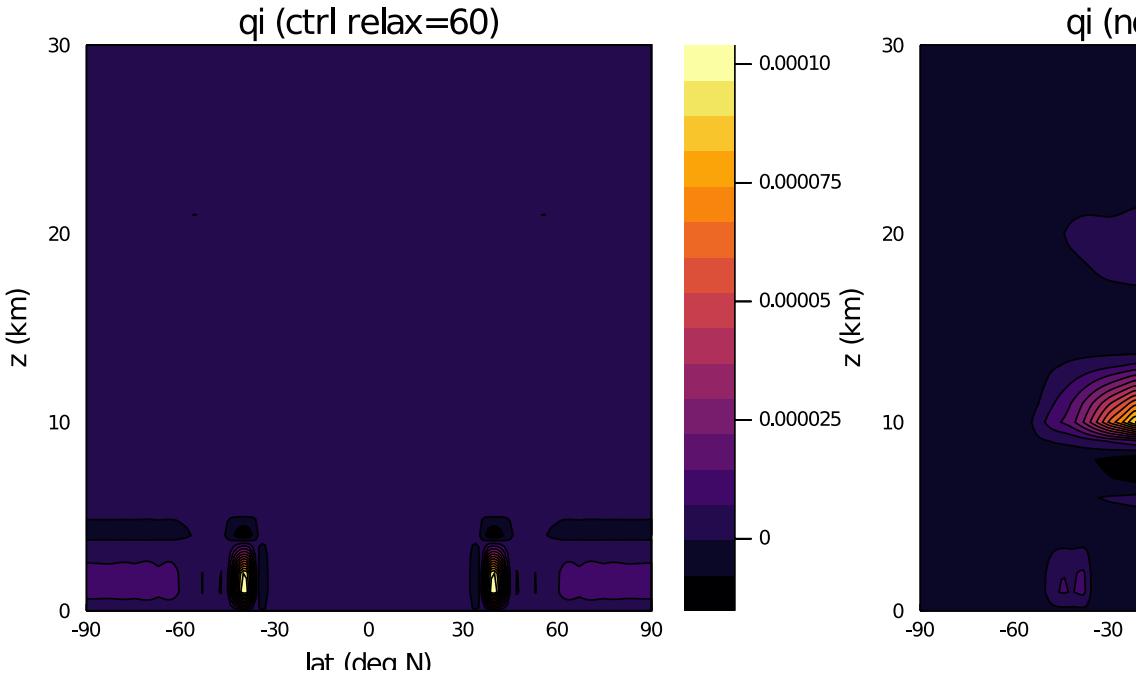
In [36]:

```
# get_zonal_mean: qi
t_slice = 3
var_name = "qi"

zm_c = get_zonal_mean(filename, var_name, nan_fill_value)
p_ctrl = contourf( lat, lev, (zm_c[:,:,t_slice])' , title="$var_name (ctrl re

# Make anomaly plots and save them in an array
plot_array = Any[p_ctrl]; # can type this more strictly
for i in 2:nexp[1]
  zm = get_zonal_mean( "$CLIMA_NETCDF/"fnames[i], var_name, nan_fill_value);
  exp_name = get_short_expname(fnames[i], "relax60_")
  one_plot = contourf( lat, lev, (zm[:,:,t_slice]-zm_c[:,:,t_slice])', title=
  push!(plot_array,one_plot); # make a plot and add it to the plot_array
end

fig=plot(plot_array... , layout=(1, 2), size=(1000, 400) )
#savefig(fig, string("$CLIMA_ANALYSIS/plot_$var_name","_hovmoller_sens.pdf"))
display(fig)
```



In [ ]:

```
1
```