CREATE AN ADVANCED TROPICAL CYCLONE HAZARD

A climada additional module

Version April 2014

David Bresch, *david.bresch@gmail.com*

Lea Mueller, *muellele@gmail.com*

Contents

The climada tropical cyclone hazard module forms an integral part of the probabilistic damage model. The hazard module characterizes the peril through the description where, how often and with what intensity events occur. The hazard module can be developed for almost any hazard (wind, flood, storm surge, landslides …). Here, we describe how to generate an **advanced** tropical cyclone hazard.

Tropical cyclones are rapidly-rotating storm systems that are characterized by a low-pressure center, strong winds and spiral arrangement of thunderstorms that produce heavy rains. Depending on their location and strength, tropical cyclones are referred differently, such as hurricanes in the western North Atlantic, eastern North Pacific, Caribbean Sea and Gulf of Mexico, typhoons in the western North Pacific, cyclones in the South Pacific, Indian Ocean, the Bay of Bengal and the Arabian Sea. Figure 1 displays historical tropical cyclones worldwide.

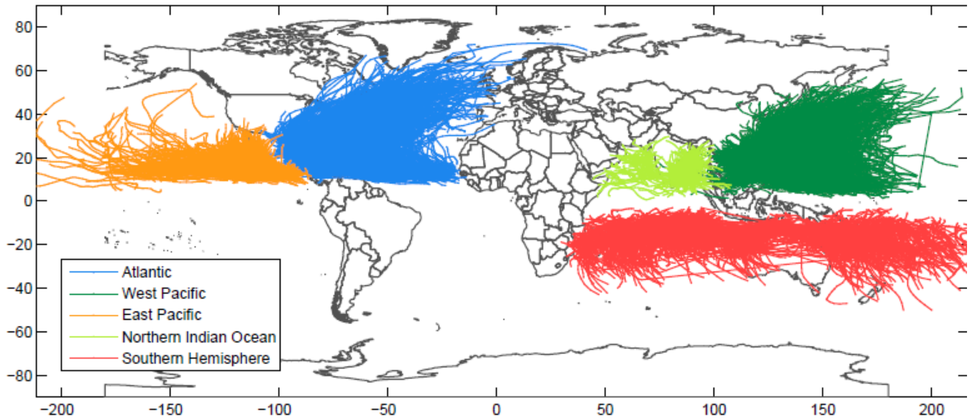


Figure 1: Historical tropical cyclone tracks in the Atlantic, the West Pacific, East Pacific, Northern Indian Ocean and the Southern Hemisphere.

1. Historical data

1a) Download the data

Download the raw data from *http://weather.unisys.com/hurricane*

Basin	*Internet page*
North Atlantic (atl)	*http://weather.unisys.com/hurricane/atlantic/tracks.atl*
East Pacific (epa)	*http://weather.unisys.com/hurricane/e_pacific/tracks.epa*
West Pacific (wpa)	*http://weather.unisys.com/hurricane/w_pacific/tracks.wpa*
Southern Hemisphere: South Indian Ocean and South Pacific (she)	*http://weather.unisys.com/hurricane/s_indian/tracks.she*
North Indian Ocean (nio)	*http://weather.unisys.com/hurricane/n_indian/tracks.nio*

Upon download, it is worth adding the extension .txt, so the files can be read with any editor (they are ASCII), such that the North Atlantic file becomes e.g. **`tracks.atl.txt`**

In order of testing routines and understanding the methodology in a faster way, a shortened version of the North Atlantic dataset, **`TEST_tracks.atl.txt`** is often used, referred to as **`TEST atl`** data (just comprising the last few years of data). You find this dataset at *…\climada\data\tc_tracks\TEST_tracks.atl.txt*

Please consult Appendix B for details of the raw data file structure.

1b) Add storm category

Add storm category according to Saffir-Simpson Hurricane Scale to every track (Table 1).

MATLAB call:

```
tc_track =    climada_tc_stormcategory(tc_track);
```

Table 1: Saffir-Simpson Hurricane Scale.

Category	*Maximum Sustained Wind Speed**				*Storm Surge*
	knots	*mph*	*km/h*	*m/s*	*m*
Tropical Depression	< 34	< 39	< 63	< 18	≈ 0
Tropical Storm	< 64	< 73	< 118	< 33	0,1–1,1
Hurrican Category 1	< 83	< 95	< 153	< 43	1,2–1,6
Hurrican Category 2	< 96	<110	< 177	< 49	1,7–2,5
Hurrican Category 3	< 113	<130	< 209	< 58	2,6–3,8
Hurrican Category 4	< 135	<155	< 249	< 69	3,9–5,5
Hurrican Category 5	> 135	> 155	> 250	> 70	> 5,5

* 1-minute period, at 10 m (33 ft) above the surface

Source:
http://en.wikipedia.org/wiki/Saffir%E2%80%93Simpson_Hurricane_Scale

2. Probabilistic data

2a) Incorporate wind speed decay after landfall

In order to have realistic probabilistic tracks, a wind speed decay function needs to be incorporated. The wind speed decay function can be derived from the historical tc tracks. We use an exponential decay function as proxy for the wind speed decay. The exponential decay function is calculated with the function **`climada_tc_track_wind_decay_calculate`** y and the parameters are saved in the file **`p_rel.mat`**. For more information on wind decay after landfall see Appendix 6a).

Firstly we need to identify the timing of landfalls to each tc track structure and hence we add an on land variable for every tc track. Hint: see **`border_mask`** (matlab variable) to understand land/sea differentiation.

MATLAB call:

```
tc_track =        climada_tc_track_on_land (tc_track, border_mask)
```

Secondly, we calculate the wind speed decay after landfall based on historical tc tracks

MATLAB call:

```
[tc_track         climada_tc_track_wind_decay_calculate(tc_track,check_plot);
p_rel] =
```

And thirdly, we apply the wind speed decay function to the probabilistic tracks

MATLAB call:
```
tc_track_prob =   climada_tc_track_wind_decay(tc_track_prob,p_rel,
                  check_plot);
```

Finally add the saffir simpson storm category to each tc track based on the maximum sustained wind speed over the track lifetime.

MATLAB call:
```
tc_track =    climada_tc_stormcategory (tc_track);
```

Absolute wind speed decay

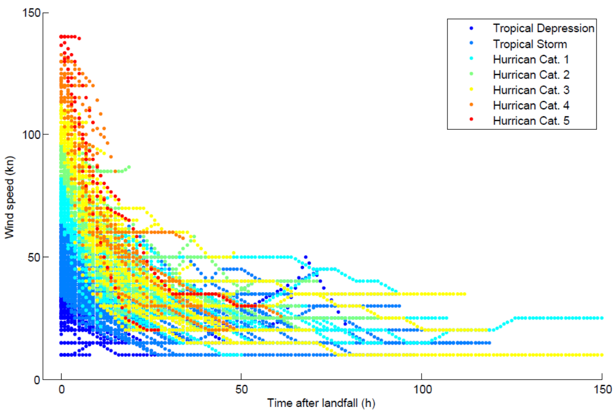


Figure 2: Absolute wind speed decay after landfall of historical tracks in the North Atlantic (1445 tracks).

Relative wind speed decay

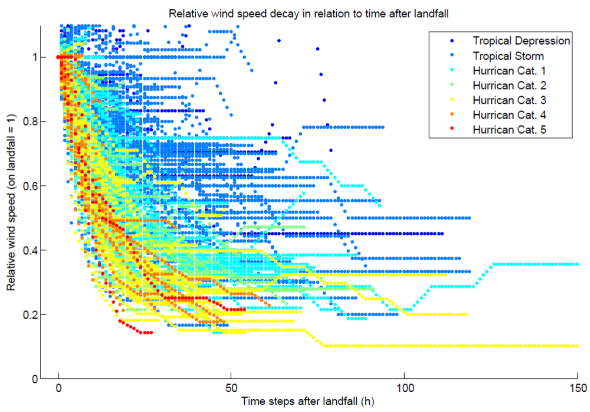


Figure 3: Relative wind speed decay after landfall of historical tracks in the North Atlantic (1445 tracks). Wind at landfall equals 1.

Relative wind speed decay with fitted functions

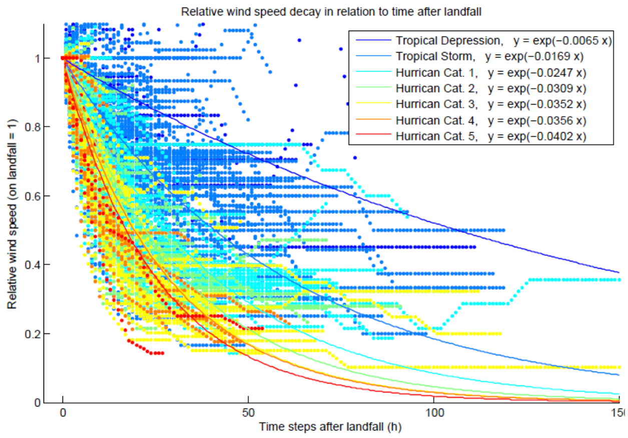


Figure 4: Relative wind speed decay after landfall of historical tracks in the North Atlantic (1445 tracks). Wind at landfall equals 1. The lines are the fitted functions for the exponential decay with starting point 1 at landfall.

Fitted functions of wind speed decay

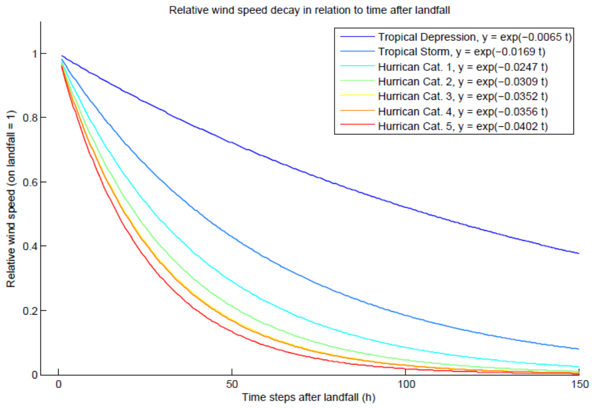


Figure 5; Relative wind decay, fitted functions for tropical depression, storm and Hurricanes category 1 to 5.

3. Hazard

3a) Calculate distance to coast for centroids

Calculate distance to closest coastline for every centroid. The basis for this calculation is the coastline data. The coastline data is downloaded from NOAA[1] and saved in *climada_additional\modul\data\coastline.txt*.

MATLAB call:

```
coastline = climada_coastline_read (centroids, coastline_file,
            check_plot)
```

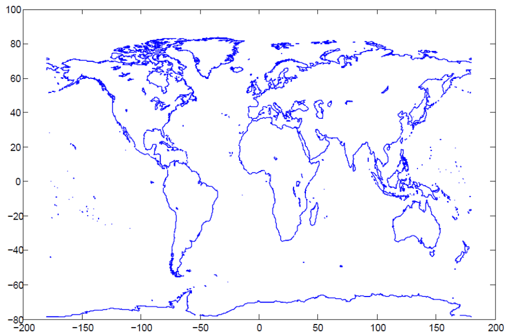


Figure 6: Coast line downloaded from NOAA.

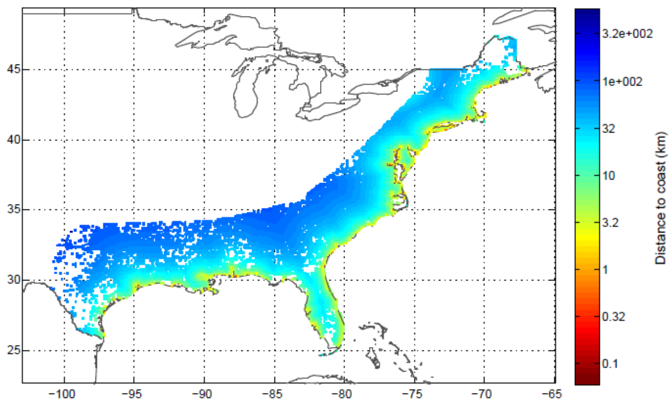


Figure 7: Distance to coast for every centroid in km and displayed with a logarithmic colour scheme.

[1] Coastline data from NOAA *http://www.ngdc.noaa.gov/mgg/coast/*

MATLAB call:

```
centroids =   climada_centroids_distance_to_coast(centroids,
              coastline, check_figure)
```

1x 1 struct

.excel_file_name	*array*
.centroid_ID	
.Longitude	
.Latitude	*vector (No. of centroids x 1)*
.VALUE	
.names	
. city_ID	*only for Mozambique*
.onLand	*1: on land, 2: buffer, 0: sea*
.dist_to_coast	*in km*

3b) Generate the wind footprint(s)

Generate wind field resulting from single track of tropical cyclone. The function converts **`tc_track.MaxSustainedWind`** in knots to **`res.gust`** in m/s.

Normally the wind footprint calculation is tested on a single tc track prior to generation of the hazard event set of all the entire historical and probabilistic track set. The tc windfield calculations are speeded up by only calculating for centroids within 750 km distance of min,max track lon/lat and by not assigning **`res.node_lon(centroid_i), res.node_lat(centroid_i).`**

Uses function **`climada_gridded_VALUE`**

MATLAB call:

```
res =   climada_tc_windfield (tc_track, centroids, equal_timestep,
        silent_mode, check_plot)
        climada_tc_windfield (tc_track, centroids, 1,1,1);
```

1x 1 struct

.gust *(wind speed in m/s at all centroids)*

.node_Azimuth

.node_lat

.node_lon *vector (1 x No. of centroids)*

.ID

.lat

.lon

Method:

Currently, the code implements the Holland windfield[2]. Given that the distance of the centroid (D) to the eye of the storm is smaller than its corresponding radius (R), the wind speed (S) is given by:

$$S = \begin{cases} \min\left(M, \quad M + 2 \cdot T \cdot \dfrac{D}{R}\right) & D \le R \quad \textit{in the inner core} \\ \max\left(0, \quad \left(\left(M - abs(T)\right) \cdot \dfrac{R^{1.5}}{D^{1.5}} \cdot e^{1-\frac{R^{1.5}}{D^{1.5}}}\right) + T\right) & D < 10 \cdot R \quad \textit{in the outer core} \\ 0 & D > R \quad \textit{out of radius} \end{cases}$$

where M denotes the maximum sustained wind and T is the celerity (forward speed). In case where D is still ten times smaller than R, you find yourself in the outer core of the storm where the wind speed takes the form of the second line in the equation above. If none of these cases are true, the wind speed is set to zero.

[2] Holland, G. J., 1980: An analytic model of the wind and pressure profiles in hurricanes. Monthly Weather Review, 108, 1212-1218.

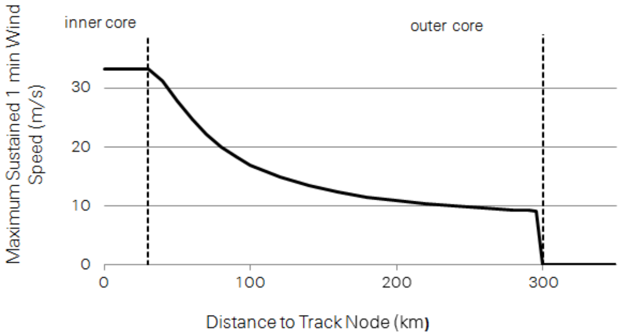


Figure 8: Maximum sustained 1 min wind speed in relation to the distance to the track node.

The radius of maximum wind (R, in km) depends on the latitude of the track node (L) as follows:

$$R = \begin{cases} 30 & L \le 24° \\ 30 + 2.5 \cdot abs(L) - 24 & L > 24° \\ 75 & L > 42° \end{cases}$$

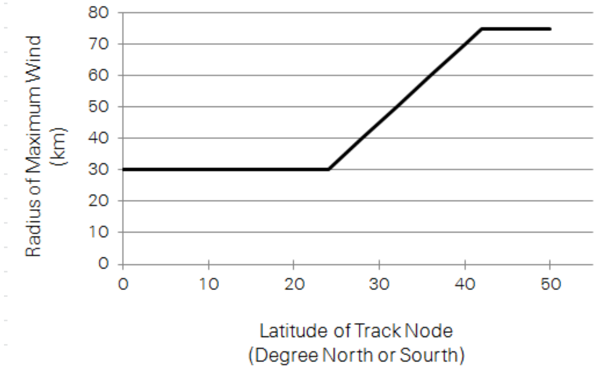


Figure 9: Radius of maximum wind in relation to latitude of track node.

Finally the wind speed (S) describes the maximum sustained 1 min wind speed. To get wind gusts that a few seconds (3-5 s) wind peaks are typically around 27% higher than a 1 min sustained wind in a hurricane environment[3].

[3] http://www.prh.noaa.gov/cphc/pages/FAQ/Winds_and_Energy.php

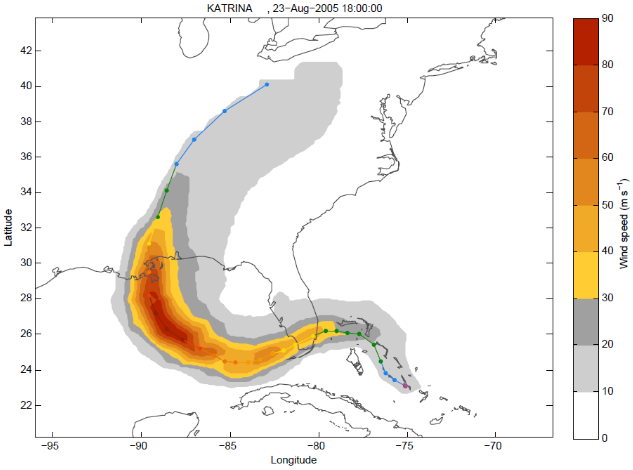


Figure 10: Wind footprint calculated based on track 1340 Katrina.

Figure saved in *...\results\footprint_NNN_1199506.pdf*

Any other windfield parametrization can be implemented in a similar fashion (just implement in a copy of climada_tc_windfield, e.g. **climada_tc_windfield2**, see also the routine **climada_tc_hazard_set** to change the caller when generating the probabilistic set).

In order to test the wind field calculation, the following might help:

Use the tc_track structure (should still be in memory), but start with only one track, e.g. tc_track(84) for the 84th track. Investigate tc_track.name to find a particular event. Use e.g. the following code to show a list of track number, year and name:

```
for i=1:length(tc_track)
     fprintf('%i %i %s\n',i,tc_track(i).yyyy(1), char(tc_track(i).name));
end
```

Load the centroids using **centroids = climada_centroids_read('',1)**, note that this call also plots the centroids (use the zoom function on the map). See also the parameter **check_plot** in the PARAMETER section of the code or refer to the routine **climada_color_plot**.

3c) Calculate the wind fields for a single track and display as animation

Refines tc track to 1hour resolution, calculates wind field for every time step of 1h. The function displays the wind fields for selected aggregated time steps, e.g. 3h, 6h, 24h. Aggregation default is 6h.

MATLAB call:

```
climada_tc_windfield_animation (tc_track(1340), centroids, aggregation, check_avi)

res = climada_tc_windfield_timestep (tc_track(1340), centroids, equal_timestep);
```

Uses functions **climada_gridded_VALUE** and **climada_tc_windfield_timestep.**

Movie saved in *...\results\windfield_animation_trackname_24h.avi*

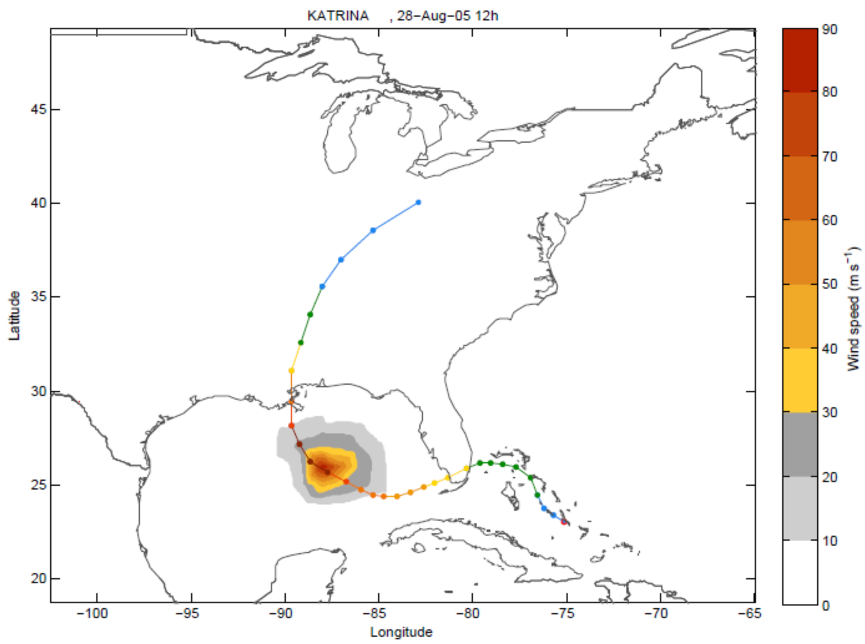


Figure 14: Wind field calculated for every timestep.

3d) Create the Hazard Set (i.e. all Footprints)

MATLAB call:

hazard = **climada_tc_hazard_set (tc_track, hazard_set_file, centroids)**

1x 1 struct	
.reference_year	
.peril_ID	
.date	*creation date of the set*
.comment	*free comment, normally containing the time the hazard event set has been generated*
.windfield_comment	
.filename	*filename of the hazard event set (if passed as a struct, this is often useful)*
.orig_years	
.event_count	
.orig_event_count	
.lon	
.lat	*vector (No. of centroids x 1)*
.centroid_ID	
.matrix_density	*density of the sparse array hazard.arr*
.event_ID	*vector (1 x No. of tc_tracks)*
.orig_event_flag	*1 for original, 0 for probabilistic*
.frequency	
.arr	***sparse array, wind field (m/s) for all storms at each centroid***

stdout for Mozambique data:

```
processing 4480 tracks (updating waitbar with estimation of time remaining every
100th track)
generating 4480 windfields took 256.157000 sec (0.057178 sec/event)
saving hazard set as C:\Documents and
Settings\s3bxxw\Desktop\Lea_climada_local\climada\data\hazards\TC_MO
```

Code:

```
for track_i=1:length(tc_track)
     res = climada_tc_windfield(tc_track(track_i),centroids,1,1);
     hazard.arr(track_i,:) = sparse(res.gust); % fill hazard array
end
```

Note the change of frequency when generating 9 probabilistic tracks of each historical one:

```
frequency = 1 / (orig_years * (ens_size + 1)) = 1 / ori_frequency / (ens_size+1)
frequency is diminished by factor 10
```

3e) Hazard weakening due to distance to coast

A correction for wind speed decrease is introduced in the **hazard.arr** array in order to incorporate the weakening of tropical cyclone when over land. The hazard weakening function depends on the distance to coast as well on the category of the tropical cyclone.

MATLAB call:

```
hazard =    climada_hazard_distance_to_coast (hazard, centroids,
            tc_track)
```

The weakening of the tropical cyclone is assumed with an exponential decay as a function of the distance D from the centroid to the coast (in km). a and b are parameters that vary depending on the cyclone intensity (Saffir-Simpson Scale). The values for the parameters a, b and c for different countries are listed in Table 1.

$$Weakening\ factor = \max\left(a \cdot e^{-D/b} \;\middle|\; c\right)$$

Table 1: Parameters for cyclone weakening function depending on cyclone category and country.

USA

Cyclone category	a	b	c
Tropical depression, storm	0.90	700	0.90
H. cat 1 and 2	0.85	500	0.85
H. cat 3, 4, and 5	0.85	320	0.35

Australia

Tropical depression, storm	1	700	0.9
H. cat 1 and 2	1	25	0.5
H. cat 3, 4, and 5	1	10.5	0.2

Japan

Tropical depression, storm	1	700	0.9
H. cat 1 and 2	1	25	0.5
H. cat 3, 4, and 5	1	10.5	0.2

China

Tropical depression, storm	1	50	0.1
H. cat 1 and 2	1	15	0.1
H. cat 3, 4, and 5	1	8.5	0.1

For more information on the calibration see Appendix 6.2 Calibration of hazard weakening6b).

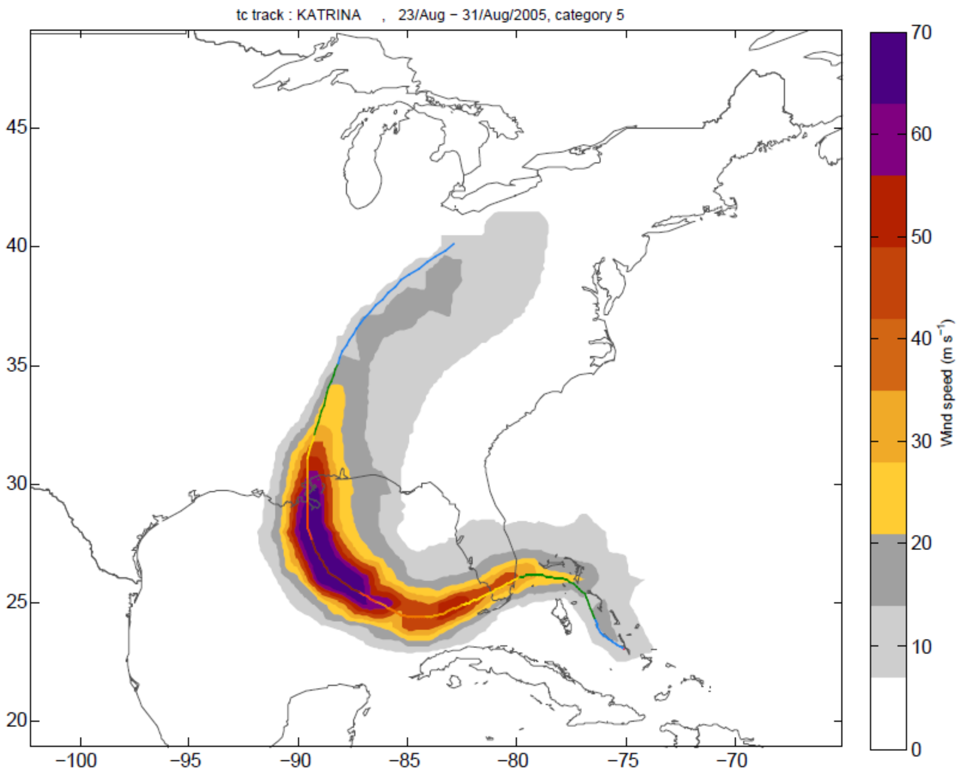


Figure 11: Unweakened hazard on the example of the footprint of Hurricane Katrina.

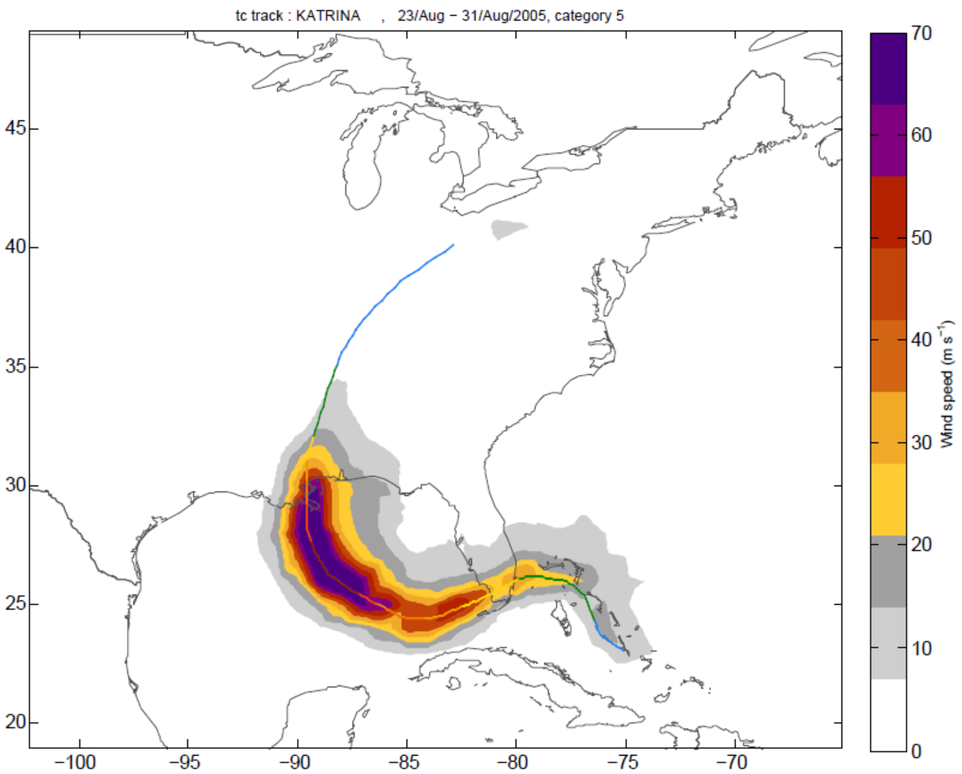


Figure 12: Weakened hazard on the example of the footprint of Hurricane Katrina.

3f) Create a climate change hazard set

The future hazard set is based on a climate change scenario and today's hazard set. The intensity or frequency of storms, or particular categories of storms, can be changed accordingly to a specific climate change scenario, by the input structure *screw*. The default values for screw are the climate change projections from IPCC special report on weather extremes (SREX, March 2012[4]) with the given time horizon of 2100, see Table 2. The time horizon can be set for any given year between today and 2100 and the climate change projections are linearly interpolated.

Table 2: Default values for screw. The climate change scenario is based on the IPCC SREX.

screw(1)	.variable_to_change	'frequency'
	.frequency	0.8
	.time_horizon	2100
	crew.cat	[4 5]
screw(2)	.variable_to_change	'frequency'
	.frequency	-0.28
	.time_horizon	2100
	.cat	[0 1 2 3]
screw(3)	.variable_to_change	'intensity'
	.frequency	0.11
	.time_horizon	2100
	.cat	[0 1 2 3 4 5]

MATLAB call:

hazard = **climada_hazard_clim_scen** (hazard, tc_track, hazard_save_name, reference_year, screw)

climada_hazard_clim_scen

1x 1 struct

.reference_year	Is the requested time horizon, e.g. 2017, 2030...
.lon	
.lat	vector (No. of centroids x 1)
.centroid_ID	
.orig_years	

[4] *http://www.ipcc-wg2.gov/SREX/images/uploads/SREX-All_FINAL.pdf*

.orig_event_count	
.event_count	
.event_ID	vector (1 x No. of tc_tracks)
.orig_event_flag	1 for original, 0 for probabilistic
.frequency	
.arr	sparse array, wind field (m/s) for all storms at each centroid, including **climate change scenario based on change in frequency and intensity**.
.matrix_density	density of the sparse array hazard.arr
.windfield_comment	
.peril_ID	
.filename	filename of the hazard event set (if passed as a struct, this is often useful) and enhanced with _cc_reference year
.comment	TCNA climate change scenario
.date	creation date of the set

stdout

```
Reference year for hazard_cc: 2017
***
frequency increased by       4.55% for category 4  5 for reference year 2017
***
frequency decreased by      -1.59% for category 0  1  2  3 for reference year 2017
***
intensity increased by       0.63% for category 0  1  2  3  4  5 for reference year
2017

***Climate change scenario ***
  saved in C:\Documents and saved in C:\Documents and
Settings\s3bxxw\Desktop\Lea_climada_local\climada\data\hazards\hazard_clim.mat
```

3g) Analyze Statistics; Plot Wind Speed for Specific Return Periods at all Centroids for Historical Data Set, Probabilistic Data Set or Climate Change Scenario

Plot wind speed based historical, probabilistic or climate change data, for requested return periods at all centroids. If no return periods are specified, it takes return periods indicated in **climada_global.LFC_return_periods**.

MATLAB call:

```
hazard = climada_hazard_stats (hazard, return_periods,
         hazard_R_file, check_plot, centroids, rain,
         check_printplot)
         climada_hazard_stats ([], 1)
```

.intensity_fit_ori

.R_fit_ori

.intensity_fit

.R_fit

Figures can be saved in

...\results\hazard_stats_historical.pdf

...\results\hazard_stats_probabilisitc.pdf

...\results\hazard_stats_climate.pdf

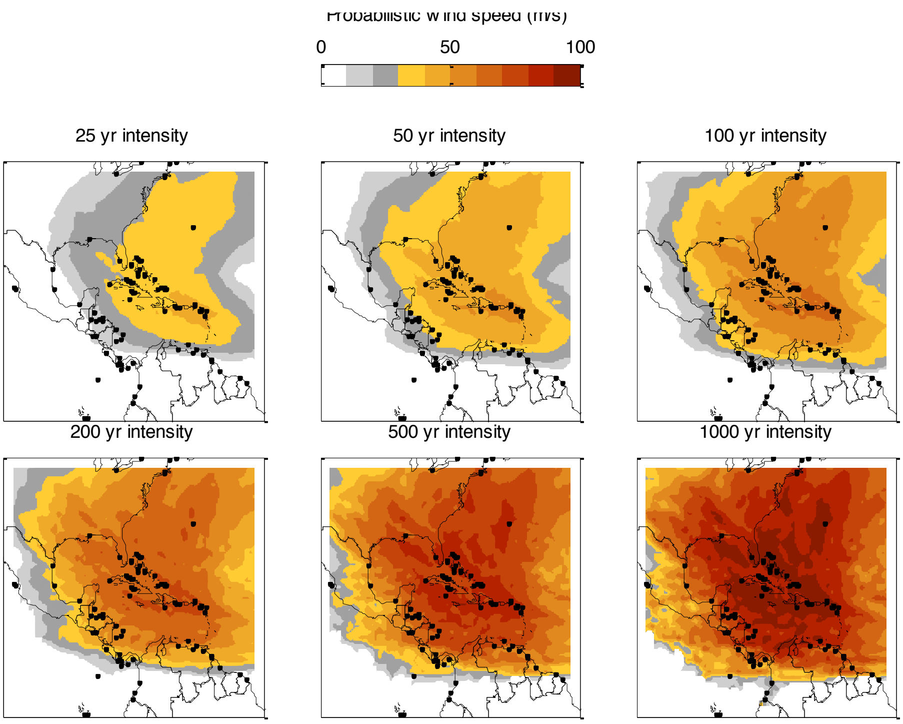


Figure 15: Wind speed maps for specific return periods.

4. Damage

4a) Read entity; assets, vulnerability, deductibles and covers

MATLAB call:

```
entity =    climada_entity_read (entity_filename, hazard)
            climada_entity_read
```

1x 1 struct

.assets	*.excel_file_name*	
	.Latitude	
	.Longitude	
	.Value	
	.Deductible	*vector (1 x No. of centroids)*
	.Cover	
	.VulnCurveID	
	.centroid_index	
	.Value_2030	
	.hazard.comment	
.vulnerability		
	.exce_file_name	
	.VulnCurveID (1, 2 and 3)	
	.Intensity (wind speed in m/s)	
	.MDD	*vector (1 x 30)*
	.PAA	
	.MDR	
.measures	*.excel_file_name*	
	.name	
	.color	
	.cost	
	.hazard_intensity_impact	
	.hazard_high_frequency_cutoff	
	.vuln_MDD_impact_a	
	.vuln_MDD_impact_b	
	.vuln_PAA_impact_a	
	.vuln_PAA_impact_b	
	.vuln_map	
	.risk_transfer_attachement	
	.risk_transfer_cover	
	.color_RGB	

	.vuln_mapping
.discount	.excel_file_name
	.yield_ID
	.year
	.discount_rate

stdoutput

```
entity saved as mat-file in
...\climada\data\entities\entity_USFL_MiamiDadeBrowardPalmBeach2010.mat
```

4b) Calculate the Event Loss Set (ELS)

Compute event loss for every storm and sum up for all centroids, based on asset value, MDD, PAA, deductibles and cover.

MATLAB call:

```
ELS =     climada_ELS_calc (entity, hazard, annotation_name, ELS_save_file)
          climada_ELS_calc
```

1x 1 struct

.hazard	*.filename*	
	.comment	
.assets	*.filename*	
.vulnerability	*.filename*	
.annotation_name		
.comment		
.event_ID		
.frequency		*vector (No. of storms x 1)*
.orig_event_flag		
.loss		
.reference_year		*2010*
.Value		*total value of assets*
.EL		*losses for all storms summed up per centroid*
.loss_sort		
.loss_ori_sort		
.R		
.R_ori		

Fitted wind speed for specific return periods

Linear interpolation between loss points

.loss_fit_ori

.loss_fit

.R_fit_ori

.R_fit

stdout

```
processing 189 assets and 1379 events, calculation took 0.188000 sec (0.000995
sec/event)
saving ELS as ...\climada\data\results\ELS_2030_clim.mat
```

Technical Hints

The variables have speaking names, note that the inner loop is vectorized. The key code within **climada_ELS_calc** (about line 155ff):

```
for asset_i=1:n_assets
     temp_loss = entity.assets.Value(asset_i)*MDD.*PAA
end % asset_i
```

with:

- temp_loss since it will be added in an 'outer loop' over asset_i
- entity.assets.Value(asset_i) is the Value of asset_i
  entity is a structure which contains all asset and vulnerability data
- MDD is here a vector of MDDs, PAA is the vector or PAAs
- .* is the element-wise (scalar) multiplication
- ELS stands for Event Loss Set, the set (or vector) of all event losses

In that calculation, the hazard intensity did not show up in the calculation, did we miss something? Well, the vulnerability is a function of the hazard intensity, hence:

```
MDD = f(hazard intensity)
PAA = f(hazard intensity)
```

where hazard intensity is the hazard intensity at asset_i for event_j, but event_j never shows up in the code, since the code is vectorized along the event dimension for performance reasons

And now, it gets technical (no way around this, sorry) – how to get the vector of MDDs

Remember: outer loop (explicit) over assets, inner loop (implicit) over events:

```
% approx line 115 in climada_ELS_calc.m
for asset_i=1:n_assets
    % the index of the centroid for given asset in the hazard
    % set     asset_hazard_pos = entity.assets.centroid_index(asset_i);

% find the vulnerability for the asset under consideration
    asset_vuln_pos        =        …find(entity.vulnerability.VulnCurveID        ==
     entity.assets.VulnCurveID(asset_i));
% convert hazard intensity into MDD: we need a trick to      % apply interp1 to the
SPARSE hazard matrix: we evaluate only at non-zero elements, therefore need a
function handle (the @ % below) to pass vulnerability to climada_sparse_interp:
interp_x_table= entity.vulnerability.Intensity(asset_vuln_pos);
interp_y_table = entity.vulnerability.MDD(asset_vuln_pos);

% apply to non-zero elements only
     MDD = spfun (@climada_sparse_interp,
            hazard.arr(:,asset_hazard_pos));

% similarly, convert hazard intensity into PAA
    interp_y_table = entity.vulnerability.PAA(asset_vuln_pos);

PAA = spfun (@climada_sparse_interp,
           hazard.arr(:,asset_hazard_pos));
    % calculate the from ground up (fgu) loss
    temp_loss = entity.assets.Value(asset_i)*MDD.*PAA;
```

Remember: outer loop (explicit) over assets, inner loop (implicit) over events, now, we need to sum up over assets:

```
% add to the ELS
ELS.loss  = ELS.loss  + temp_loss';
% add Value
```

```
ELS.Value = ELS.Value + entity.assets.Value(asset_i);
end % asset_i
```

A note on ' : for historical reasons the ELS.loss vector is transposed

Technical Hints on Insurance Conditions

Remember: outer loop (explicit) over assets, inner loop (implicit) over events

```
% approx line 115 in climada_ELS_calc.m
for asset_i=1:n_assets
    […]
    % calculate the from ground up (fgu) loss
    temp_loss = entity.assets.Value(asset_i)*MDD.*PAA;
if entity.assets.Deductible(asset_i)>0 ||
entity.assets.Cover(asset_i) <
           entity.assets.Value(asset_i)

        % apply Deductible and Cover
        temp_loss = min(max(temp_loss-...
            entity.assets.Deductible(asset_i)*PAA,0),...
            entity.assets.Cover(asset_i));
    end
[…]
end % asset_i
```

and similar for any conditions on the event loss set (ELS), always of the form min(max(loss-deductible,0) ,cover).

6. Appendix

6a) Background on hurricane wind speed decay after landfall

NOAA, Hurricane Research Division[5],

The sustained (1 min or longer average) winds are reduced because of the dampening effect of larger roughness over land (i.e. bushes, trees and houses over land versus a relatively smooth ocean). The gusts are stronger because turbulence increases and acts to bring faster winds down to the surface in short (a few seconds) bursts.

After just a few hours, a tropical cyclone over land will begin to weaken rapidly - not because of friction - but because the storm lacks the moisture and heat sources that the ocean provided. This depletion of moisture and heat hurts the tropical cyclone's ability to produce thunderstorms near the storm centre. Without this convection, the storm rapidly fills.

Nature conducted this experiment during Andrew as the hurricane traversed the very wet Everglades, Big Cypress and Corkscrew Swamp areas of southwest Florida. Andrew weakened dramatically: peak winds **decreased about 33%** and the sea level pressure in the eye rose 19 mb[6].

Hurricane Science, University of Rhode Island (URI), Graduate School of Oceanography (GSO)[7]

Landfall usually causes a hurricane to quickly decay. Hurricanes require evaporation from the warm ocean surface to survive. Once a hurricane makes landfall, it is separated from its ocean energy source, and hence, can no longer extract heat from the ocean. Since the air masses over land are drier and contain more aerosol particles than over the ocean, less moisture is carried into the storm, cloud coverage lessens, and air is cooled and then sinks, disrupting the hurricane's secondary circulation and hindering critical thunderstorm development. To a lesser extent, the increased roughness of the land surface also weakens a hurricane as increased friction causes a reduction in surface circulation.

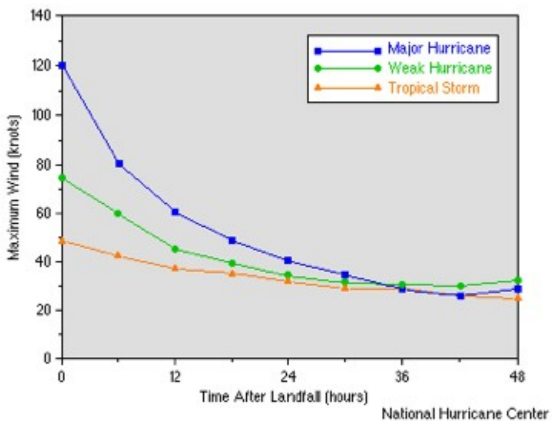


Figure 13: This graph shows how rapidly wind speed decreases once a hurricane reaches land. The roughness of the land terrain increases friction, but more critical, once over land, the system is cut off from its heat and moisture sources. Sustained winds in a hurricane will decrease at a relatively constant rate (approximately half

---

[5] NOAA Hurricane Research Division: *http://www.aoml.noaa.gov/hrd/tcfaq/C2.html*

[6] Powell and Houston 1996 : *http://www.aoml.noaa.gov/hrd/tcfaq/tcfaqREF.html#P*

[7] Hurricane Science, University of Rhode Island (URI):

http://www.hurricanescience.org/science/science/hurricanedecay/

the wind speed in the first 24 hours). Thus, the faster the forward speed of a landfalling hurricane, the further inland hurricane force winds may penetrate. Image credit the National Hurricane Center (NHC).

6b) Calibration of hazard weakening

USA

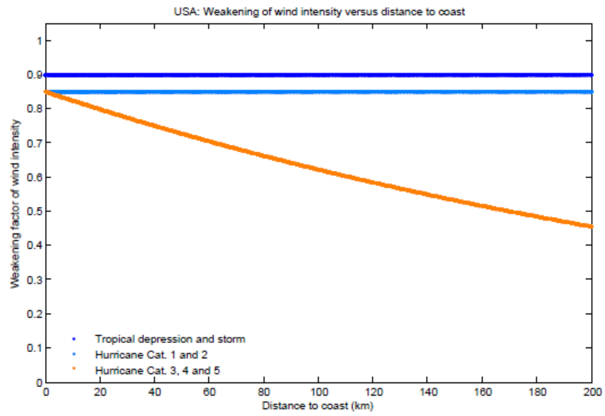


Figure 14: Hazard wind weakening depending on distance to coast and cyclone category for USA.

Australia

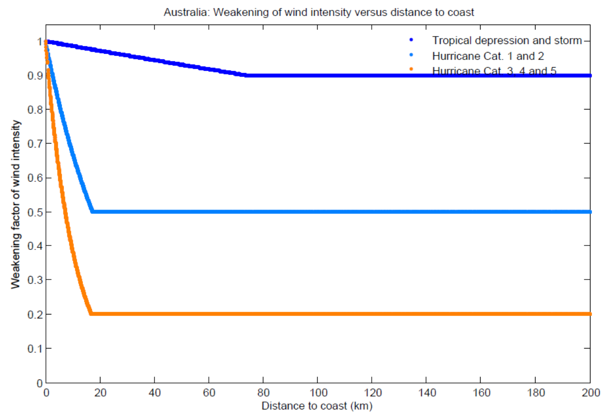


Figure 15: Wind weakening for Australia

China

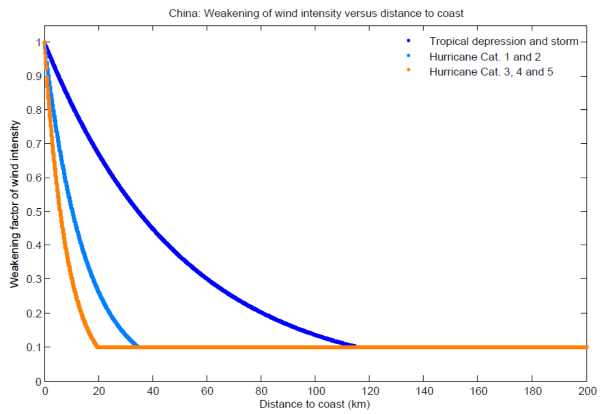


Figure 16: Wind weakening for China.

Japan

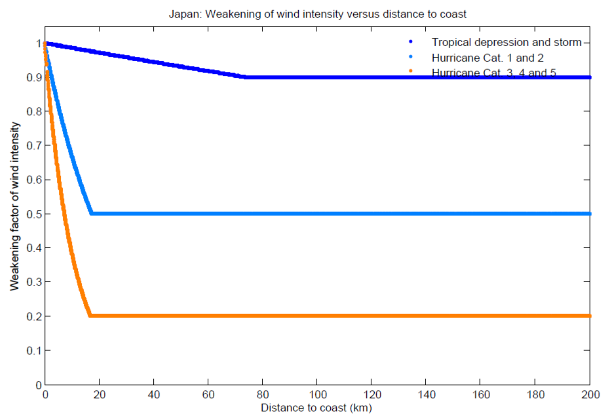


Figure 17: Wind weakening for Japan.

Comparison with MultiSNAP results

The portfolio for the calibration is the same in MultiSNAP as in climada.

USA

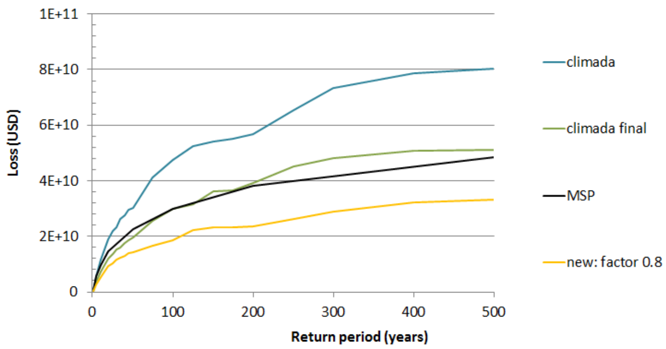


Figure 18: Hazard wind weakening depending on distance to coast and cyclone category for USA.

Australia

Figure 19: Hazard wind weakening depending on distance to coast and cyclone category for China.

China

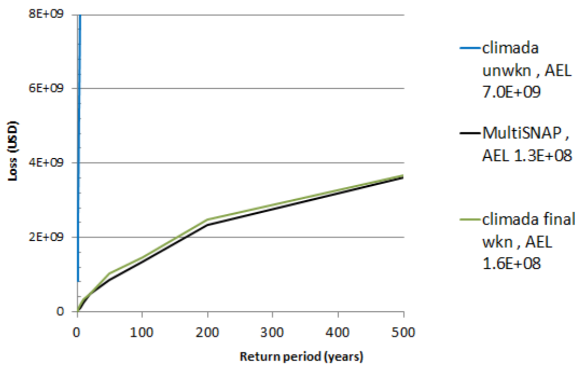


Figure 20: Hazard wind weakening depending on distance to coast and cyclone category for China.

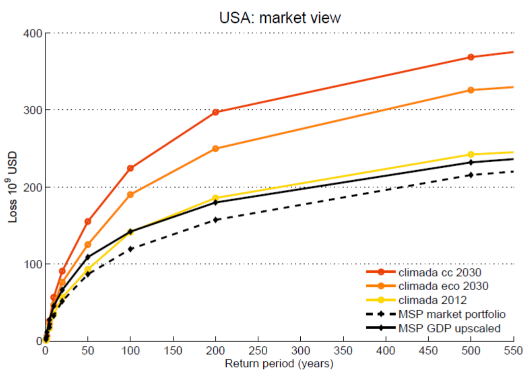
Japan

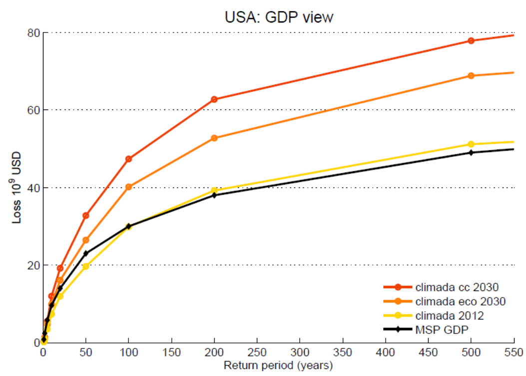
Figure 21: Wind weakening for Japan.

6c) TC global Results

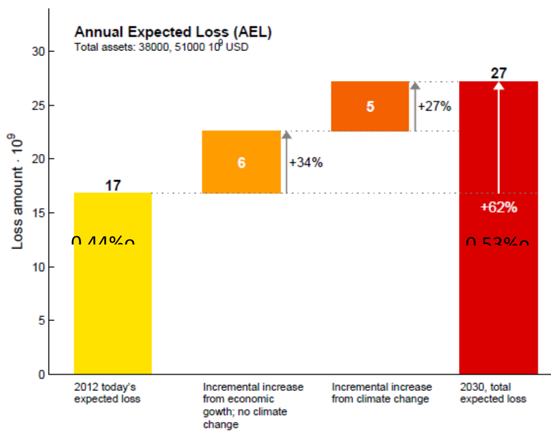
USA

Market view	GDP portfolio





AEL	AEL



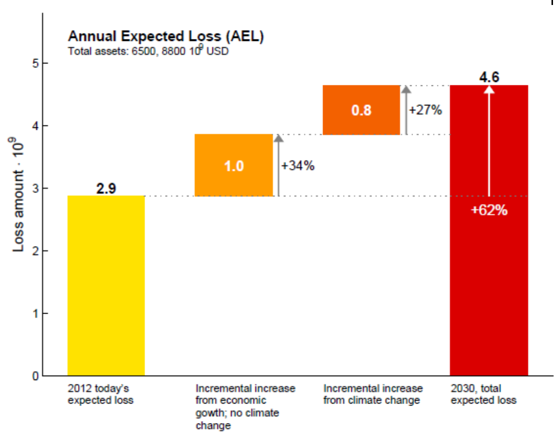
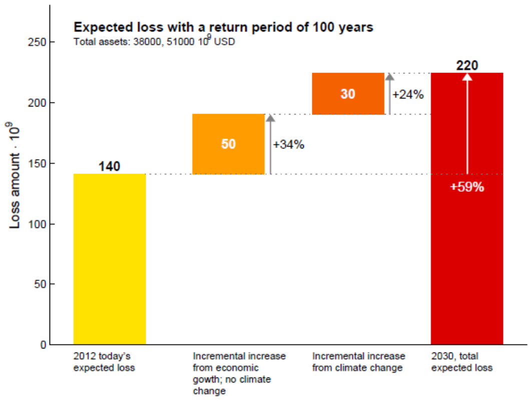
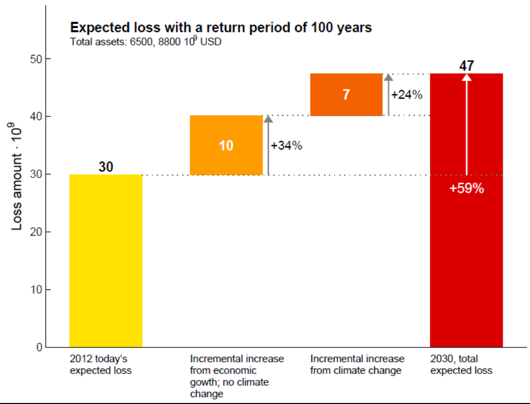


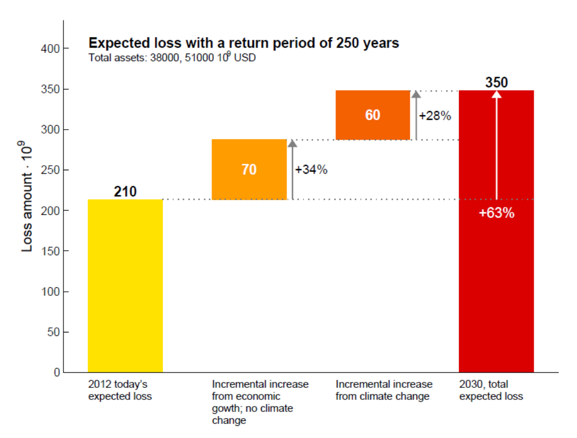
Figure 22: Risk today and in the future on the basis for annual expected loss in the USA related to tropical cyclone risk.

100 years	100 years

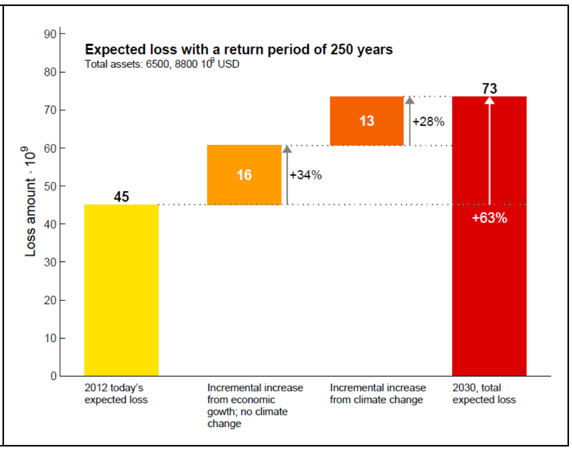




250 years	250 years



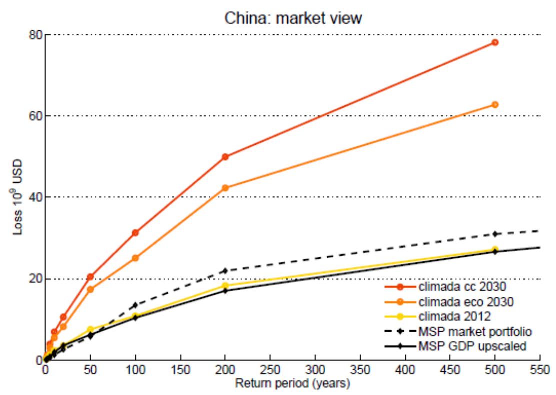
Expected loss with a return period of 250 years
Total assets: 38000, 51000 10^9 USD
Loss amount · 10^9
210
70
+34%
60
+28%
350
+63%
2012 today's expected loss
Incremental increase from economic gowth; no climate change
Incremental increase from climate change
2030, total expected loss

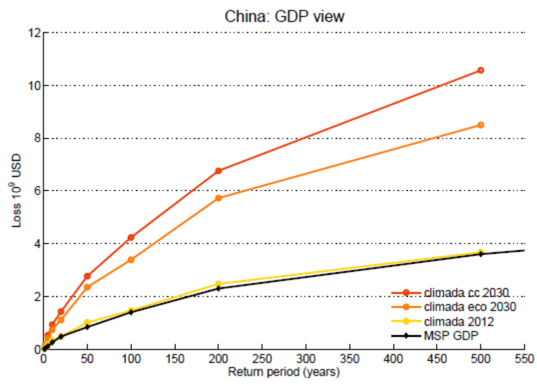


Expected loss with a return period of 250 years
Total assets: 6500, 8800 10^9 USD
Loss amount · 10^9
45
16
+34%
13
+28%
73
+63%
2012 today's expected loss
Incremental increase from economic gowth; no climate change
Incremental increase from climate change
2030, total expected loss

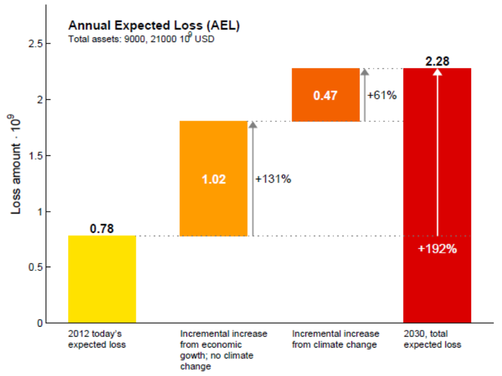
China

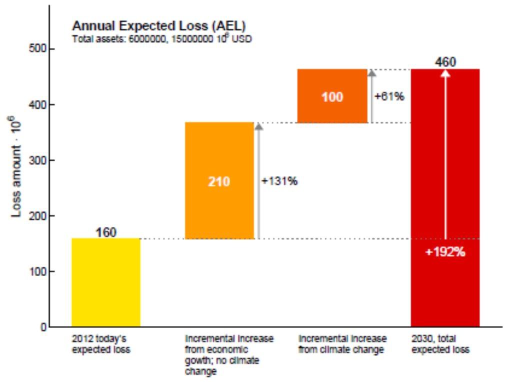
Market view	GDP portfolio



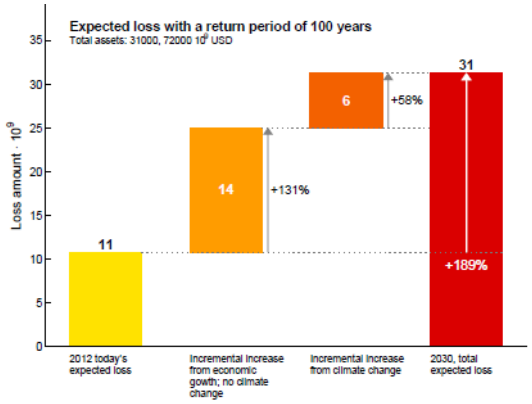


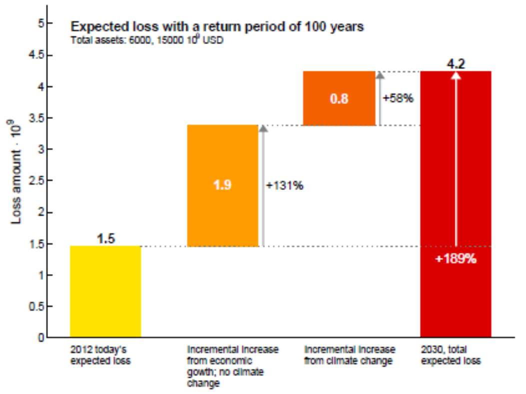
AEL	AEL



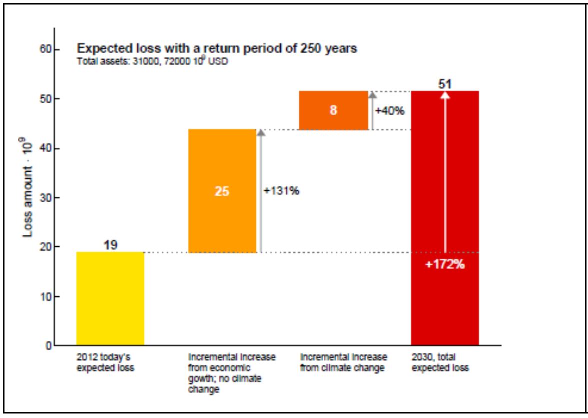


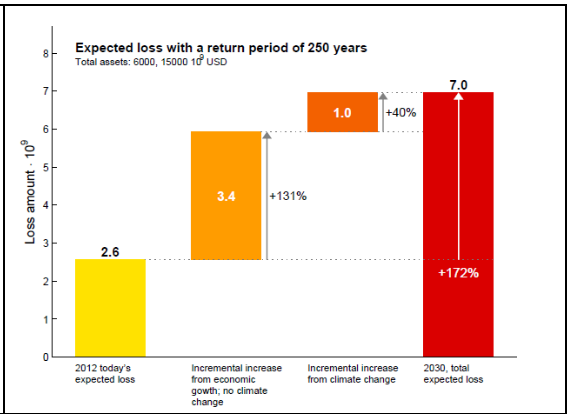
100 years	100 years





250 years	250 years


Expected loss with a return period of 250 years
Total assets: 31000, 72000 10^9 USD
Loss amount · 10^9
19
25
+131%
8
+40%
51
+172%
2012 today's expected loss
Incremental increase from economic gowth; no climate change
Incremental increase from climate change
2030, total expected loss


Expected loss with a return period of 250 years
Total assets: 6000, 15000 10^9 USD
Loss amount · 10^9
2.6
3.4
+131%
1.0
+40%
7.0
+172%
2012 today's expected loss
Incremental increase from economic gowth; no climate change
Incremental increase from climate change
2030, total expected loss