test_CEM

August 2, 2019

0.1 CEM + pyDeltaRCM Coupling

Testing a CEM, pyDeltaRCM coupling using the pymt framework. Initializing with a pre-evolved delta from pyDeltaRCM to see how the models interact with each other.

[1]:
```
from pymt.models import Cem
import numpy as np
import matplotlib.pyplot as plt
```

```
 models: Avulsion, Plume, Sedflux3D, Subside, FrostNumber, Ku,
Hydrotrend, Cem, Waves
```

[2]:
```
### Create class
coast = Cem()
```

[3]:
```
# initialize the model
args = coast.setup(number_of_rows=100, number_of_cols=100, grid_spacing=50,␣
 ↪shoreface_depth=0.0)
coast.initialize(*args)
```

[4]:
```
# set some values
coast.set_value('sea_surface_water_wave__height', 1.)
coast.set_value('sea_surface_water_wave__period', 7.)
coast.
 ↪set_value('sea_surface_water_wave__azimuth_angle_of_opposite_of_phase_velocity',0.
 ↪ * np.pi / 180.)
```

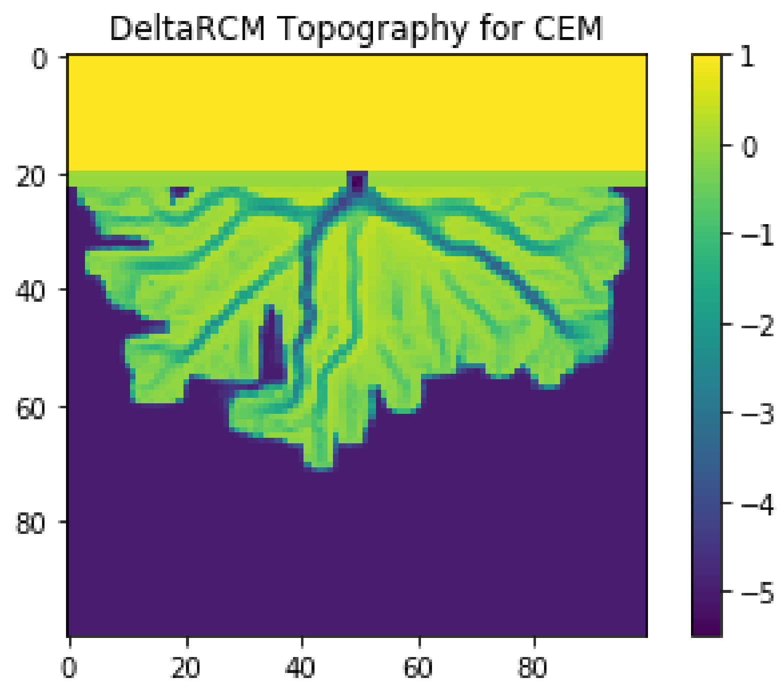
[4]:
```
array([ 0.])
```

0.1.1 Loading of the pre-evolved topography

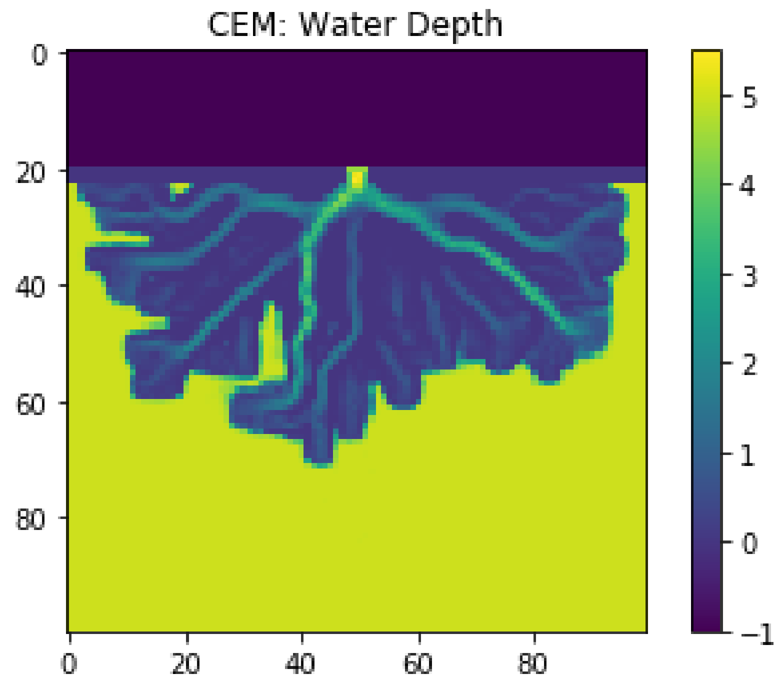
Load, visualize and then assign pre-evolved topography to CEM model.

[5]:
```
# Loading and assignment of DeltaRCM topography
eta = np.load('eta.npy')
neweta = np.empty(coast.get_grid_shape(2))
neweta[0:20,:] = 1.0
neweta[20:100,:] = eta[0:80,150:250]
```

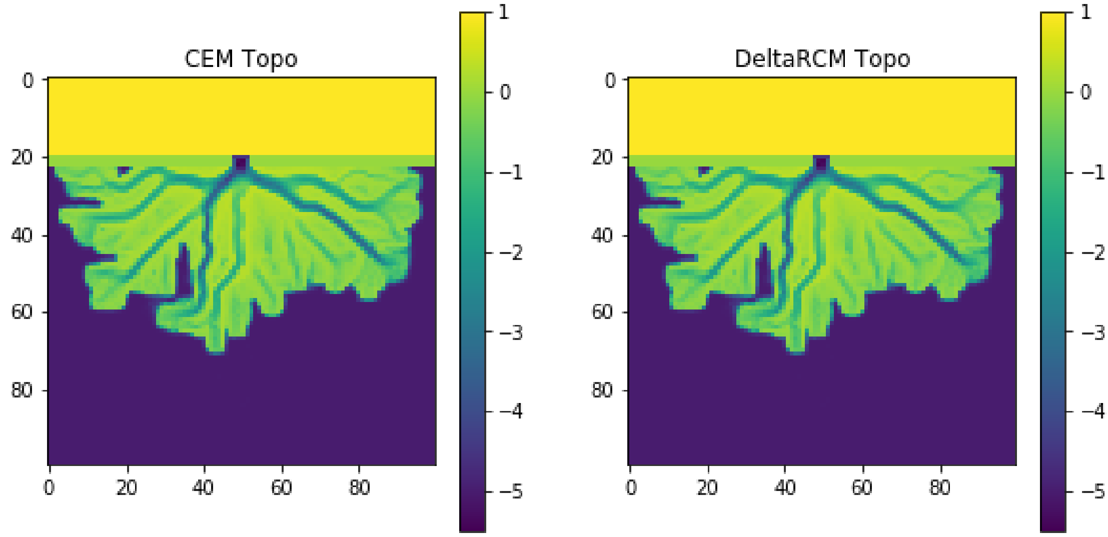
[6]:
```
plt.imshow(neweta)
plt.colorbar()
plt.title('DeltaRCM Topography for CEM')
plt.show()
```



[7]:
```
shape = coast.get_grid_shape(2)
coast.set_value('land_surface__elevation',neweta)
z = np.empty(shape)
coast.get_value('sea_water__depth',out=z)
plt.imshow(z)
plt.colorbar()
plt.title('CEM: Water Depth')
plt.show()
```

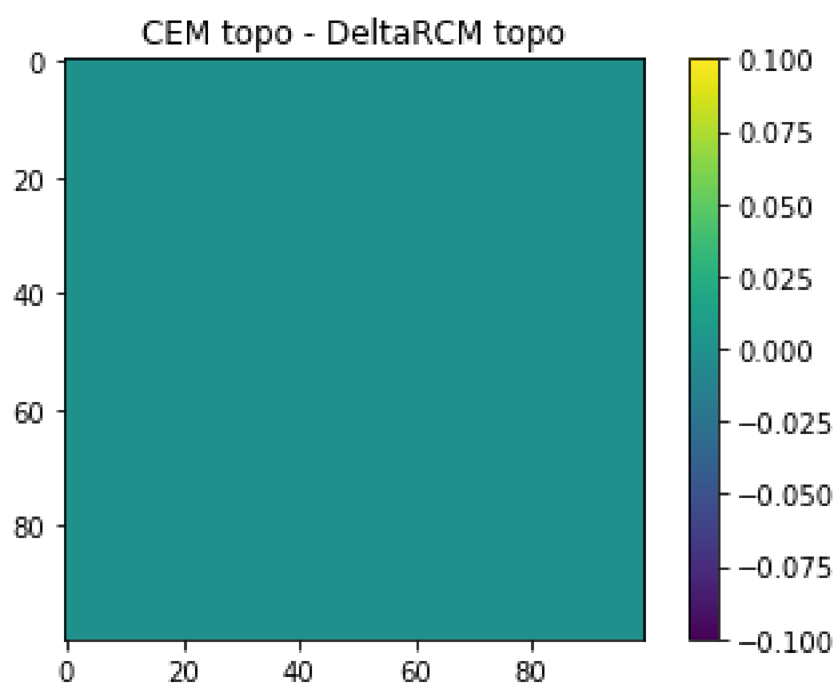


[8]:
```
z = np.empty(shape)
coast.get_value('land_surface__elevation',out=z)

plt.figure(figsize=(10,5))

plt.subplot(121)
plt.imshow(z)
plt.colorbar()
plt.title('CEM Topo')

plt.subplot(122)
plt.imshow(neweta)
plt.colorbar()
plt.title('DeltaRCM Topo')

plt.show()
```



[9]:
```
# Check to ensure the initial topography for CEM matches the DeltaRCM input␣
↪exactly
plt.imshow(z-neweta)
plt.colorbar()
plt.title('CEM topo - DeltaRCM topo')
```

[9]:
```
Text(0.5, 1.0, 'CEM topo - DeltaRCM topo')
```



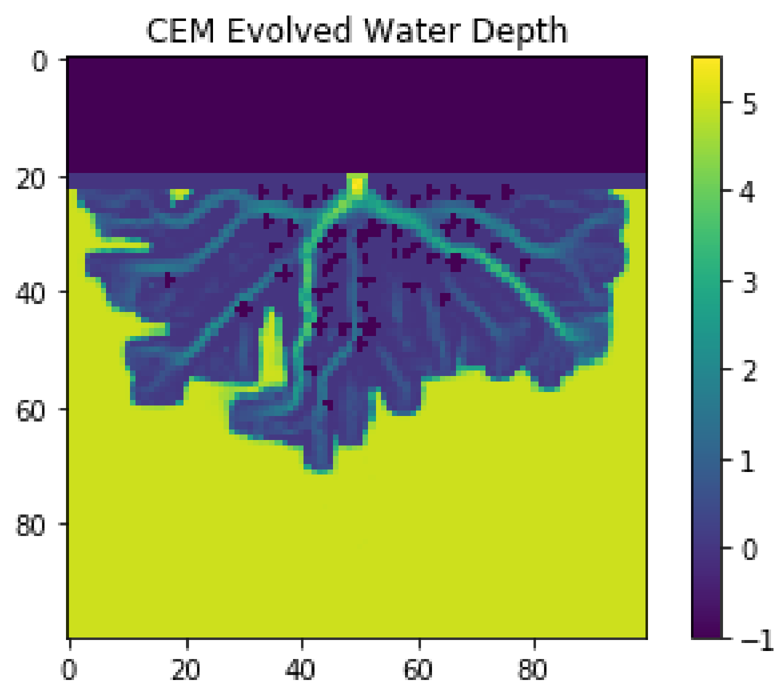
0.1.2 Run and Visualizing CEM model

Performing a single time-step of the CEM model to see what effects it has on the DeltaRCM topography established. Would like to see either some reworking/smoothing of the shoreline, or no perceptible effect as 1 timestep is not a long time for shoreline evolution.

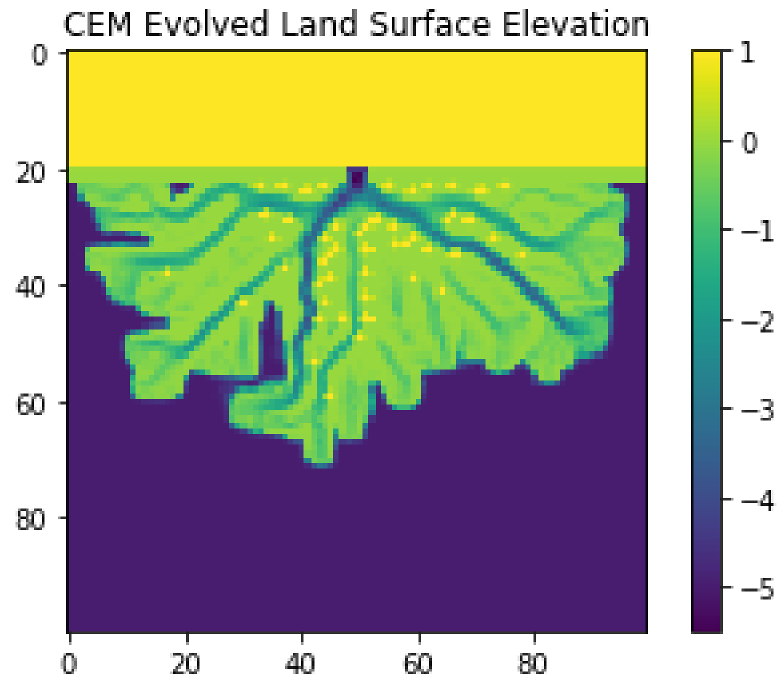
Then we view the resulting water depth and topographies from the CEM model.

[10]:
```
coast.update()
```

[11]:
```
z = np.empty(shape)
coast.get_value('sea_water__depth',out=z)
plt.imshow(z)
plt.colorbar()
plt.title('CEM Evolved Water Depth')
plt.show()
```



[12]:
```
z = np.empty(shape)
coast.get_value('land_surface__elevation',out=z)
plt.imshow(z)
plt.colorbar()
plt.title('CEM Evolved Land Surface Elevation')
plt.show()
```



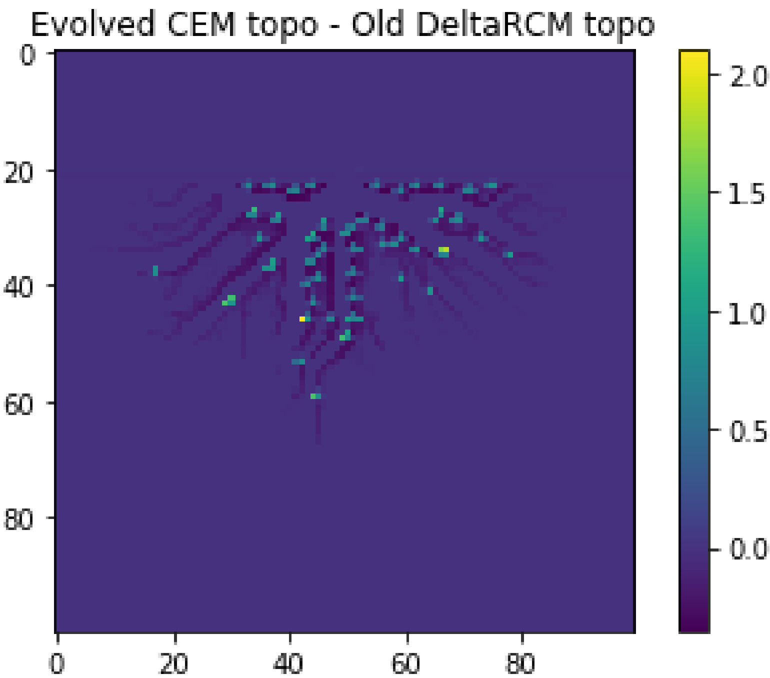
0.1.3 Topography Difference

The difference map between the old and the CEM evolved topography show reworking along the channels. This is unexpected behavior; we expect the reworking to occur primarily along the shoreline. Potential reasons for this behavior:

1. CEM expects to trace along a shoreline, the channels are being viewed as independent shorelines and so the evolution is occuring along these false shores

2. Some mistake in CEM intialization is creating unstable or unexpected conditions

[13]:
```
# Check this new topography against the initial topo from DeltaRCM
plt.imshow(z-neweta)
plt.colorbar()
plt.title('Evolved CEM topo - Old DeltaRCM topo')
plt.show()
```


Evolved CEM topo - Old DeltaRCM topo
0
20
40
60
80
0
20
40
60
80
2.0
1.5
1.0
0.5
0.0

[ ]: