The carbo* Suite of Carbonate Models

CSDMS Software Clinic, CU Boulder CO, March 2012

Chris Jenkins CU, Peter Burgess RHUL, Donald Potts UCSC

A. Clinic aims:

1. Better collaboration, effort on componentization, model validations, and applications.
2. For the guests: see 3 new approaches to carbonate modeling; perhaps apply the models in your own programs; perhaps adapt the ideas to new modeling.
3. For the hosts: get feedback from peers on the form of the modeling, possible applications, shortcomings.
4. The model codes are being made available ahead of publication. The authors reserve all copyright and priority rights on them.
5. The aim of this document is to give a recipe to clinic participants so they can pre-install the models, run them during the clinic hour, and understand the outputs.

B. General Software installation:

1. While carboCAT runs in MATLAB, carboCELL and carboLOT run under Python v2.7.
2. Installation of Python2.7 can be done using the collection of Windows 7 (64 bit) *.msi and *.exe files starting with core “python-2.7.2.msi”. We usually place Python in “c:\python27".
3. If another Python installation, or ArcGIS Python is present the installation will be more complicated. Please check this before attempting the v2.7 installation.
4. The models use these Python modules: numpy/scipy, matplotlib.
5. The modeling software, once placed in the machine will find Python through the OS.
6. If movies are to be made, install ImageMagick (supplied for clinic).

C. Running carboCELL:

1. carboCELL is a small program that plays different carbonate facies (specified as organisms) off against each other, under the complexity of external events.
2. The program folder is “carbo/carboCELL”, with Python v2.7 program “carboCELL_5-1.py”.
3. There is no environmental setup.
4. The main adjustments that can be made are through the files “facies_*.txt” and “events_*.txt” where “**” stands for the run name.
5. These run-setups are best edited from Excel (currently “facies_CJJ_3a.xls”, “events_CJJ_1a.xlsx”)

	A	B	C	D	E	F	G
	Index	Name	AccumRate_mm/yr	Solidity_/1	Longevity_yr	MaturityOnset_yr	Rivals_[Indexes]
	0	empty	0	1.00	-99	-99	[]
	1	porites	8	1.00	300	3	[4]
	2	branching acropora	10	0.03	20	1	[4]
	3	encrusting montipora	3	1.00	6	3	[3,4,5]
	4	coralline algae	2	1.00	4	1	[3]
	5	faviid	5	1.00	6	3	[3,4,5]
	6	halimeda	1	0.5	2	1	[]

a. Facies setup

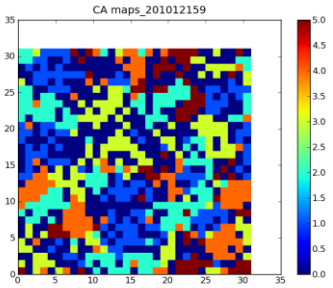
event_type	return_period_yr	mortality_/1	players_dying_[	spawning_/1	players_spawning_	time_series_URL
bleaching	7	0.3	[1,2,3,5]	0	[]	-
cots	20	0.5	[1,3,5]	0	[]	-
cyclone	14	0.1	[1,3,4,5]	1	[2]	-
disease	5	0.2	[2]	0	[]	-
sedimentation	2	0.1	[4,6]	0	[]	-
influx	3	0	[]	1	[1,2,3,4,5]	-

b. Events setup

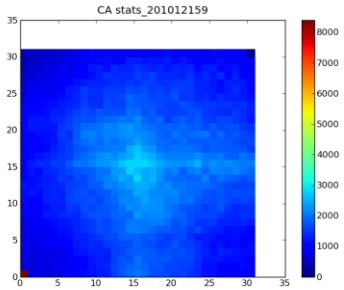
6. To run the model double-click on “carboCELL_5-1.py” or open and run it in IDLE (right click on filename). Following the run the cellular maps can be compiled into GIF movies by double clicking on “pyCarboCAT_MakeMovi_3.py”.

7. The outputs go to a subfolder which is named for the *runName* and *dateTime* (e.g., “_td/_22Mar2013_0957”). Throughout carbo* a preceding “_” (underscore) signifies that the folder or file are required by some program (not mandatory).

8. In the run outputs folder are subfolders: (i) “_dumps” – holds the tabulated output data; (ii) “_pattern” – mappings of the cellular results; (iii) “_correl” – k-spectral analyses of the maps; (iv) “_ages” – a mapping of the persistances of the cells.



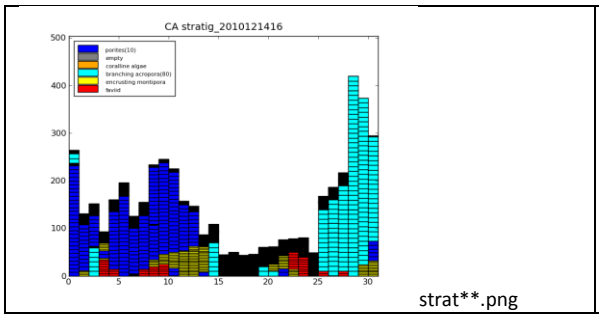
a. A map



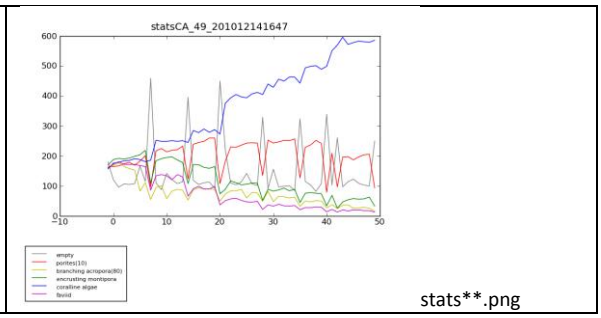
b. A spectrum

```
carboCELL run: 2011:1:22:13:3at time step: 3
[5 2 4 5 4 5 5 4 2 2 1 1 1 5 3 2 5]
[2 5 4 1 4 5 2 4 3 5 5 2 4 3 3 4 1]
[2 2 5 4 1 1 5 1 2 1 3 1 1 4 4 4 4]
[3 1 1 2 2 5 3 3 4 4 4 5 3 3 3 1 2]
[3 4 3 3 2 1 4 5 3 2 1 4 3 2 3 1 5]
[3 1 2 3 3 4 1 2 2 4 3 3 1 1 4 1 1]
[1 2 3 4 1 1 4 1 2 3 5 3 4 2 2 5 5]
[2 5 2 2 2 5 4 2 5 1 5 4 1 5 4 5 1]
[2 3 2 1 1 1 1 5 4 4 1 4 5 1 1 4]
[2 2 2 5 2 5 2 2 5 5 4 4 2 3 1 3 1]
[5 -1 2 1 3 5 3 2 1 2 2 3 4 1 2 3 1]
[1 1 3 2 1 1 3 1 1 3 2 2 2 2 3 4 2]
[2 2 1 3 5 5 5 2 5 2 1 1 5 2 3 4 5]
[5 2 5 4 4 1 4 1 1 4 3 1 3 5 5 4 4]
[2 3 1 3 4 2 1 2 4 4 2 1 2 5 2 2 5]
[5 1 3 3 1 2 3 3 3 5 3 4 2 4 2 1 1]
[4 5 5 4 2 5 5 5 4 4 3 5 4 5 5 1 4]
```

9. There are also files: in the root of the run folder: (i) “stratCA_49.png” – a cross-section of the cellular output, time on the y-axis; (ii) “statsCA_49.png” – traces of the abundance levels for each facies/organism.



strat**.png

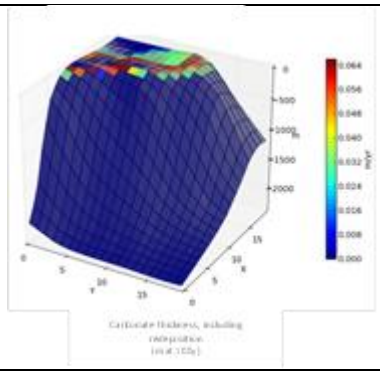


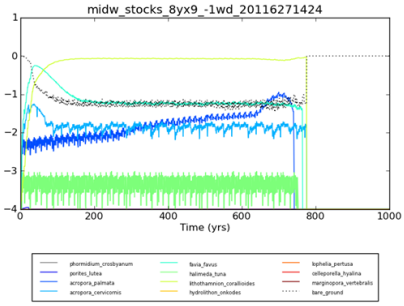
stats**.png

10. There are also files: in the root of the run folder: (i) “stratCA_49.png” – a cross-section of the cellular output, time on the y-axis; (ii) “statsCA_49.png” – traces of the abundance levels for each facies/organism, (iii) “output**movi.gif” - a movie of the cellular map outputs through time; (iv) “corr**movi.gif” – a movie of the spectral analysis outputs through time.

D. Running carboLOT:

1. carboLOT is a more involved, slower running program than carboCELL. It carries out an abbreviated multispecies Lotka-Volterra analysis as well as facies-lithology operations.
2. We will be running a pre-assembled subset of the global data that supports the carbo* suite, data on bathymetry, ocean temperature and clarity.
3. carboLOT also draws on the carboKB Knowledge Base which must first be rendered as the file “_carboKB_7.okb” (organism knowledge base) in the folder “_carboKB”.
4. The program folder is “carbo/carboLOT”, and the program is the Python v2.7 program “carboLOT_16.py”.
5. The project has two control files: (i) “_midw_SETUP.txt” – setting the locality, time base and other controls; (ii) “_midw_EVENTS.txt” – setting the severities and timings of different types of events.
6. To run carboLOT the first time for a project, select the runtime option “Use stored enviro grids (if available; else make new) ?“ to ‘NO’. For extraction the dbSEABED global data resources must be present on the machine at “c:/dbSEABED/_db9/_enviro/*”. For the clinic this is already done for several projects and the subsets held in a folder “_ENVIRO”. Those projects are “midw”, “lhis”, “agin” (Midway, Lord Howe, Agincourt reefs).
7. The outputs appear in a folder named for run time, such as “20116220837” (yyyymmddhhmm). Within each of those are the folders: (i) “_3dPlots” – facies and environmental variables on 3D topography; (ii) “_GraphPlots” – time series of organism abundances; (iv) “_MapPlots” - facies and environmental variables on 2D maps; (v) “_StockDumps” – text data on abundances, etc.; (v) “_StratDumps” – text data on the time series. There is also a run log file “_midw_logFile.txt”.





8. Once a run of carboLOT is completed there is the option of running “carboKML_2a.py” which will compose a GoogleEarth KML scene. An example is shown below. The files are produced in the root folder “carboLOT”: “_vImgModl.dae/.kml/.png”.

E. The KnowledgeBase:

1. Population ecology models typically require a large amount of input, to specify the reproductive, growth and living characteristics of species. carboLOT is not exception, though we have tried to minimize the input, maintenance and transfer operations associated with it.
2. The Python program “” serves to pick up the KB data from a *.txt copy of the XL worksheet, validate the data and write it to an intermediate format ready for carboLOT.
3. The parameters in the KB are shown below in the intermediate format.

```
#Synopsis of the Organism KB parameters -----------------
#          kbOrgnsmA[iCr]=Organism name
#
#          kbCoverageA[iCr][0]=Colony Size (m)
#          kbCoverageA[iCr][1]=Colony Frame Fraction (fxn)
#          kbCoverageA[iCr][2]=Observed Carrying Capacity (area_fxn)
#
#          kbGrowthA[iCr][0]=Colony Growth Rate (m/yr)
#
#          kbSkeletalA[iCr][0]=Disintegrated volume fraction (fxn)
#          kbSkeletalA[iCr][1]=Disintegrated grainsize (m)
#          kbSkeletalA[iCr][2]=Porosity (fxn)
#
#          kbZonesA[iCr]=Vertical 4-Zone Presences (%)
#
#          kbWdHabittA[iCr]=Habitat by WD ([m,fxn])
#          kbTempHabittA[iCr]=Habitat by Temperature ([degC,fxn])
#          kbParHabittA[iCr]=Habitat by Irradiance ([mol_photons/m2/d,fxn])
#
#          kbMortalityA[iCr][0]=Mortality (fxn per individ /yr)
#
#          kbReprodA[iCr][0][0]=Cloning Reproduction Rate (fxn/yr)
#          kbReprodA[iCr][0][0]=Cloning Reproduction Success (fxn)
#          kbReprodA[iCr][1][0]=Spawning Reproduction Rate (fxn/yr)
#          kbReprodA[iCr][1][0]=Spawning Reproduction Success (fxn)
#          (Note: units are shown as expected by the modelling)

```

4. The organisms entered in the KB are currently: *Phormidium_crosbyanum*, *Porites_lutea*, *Acropora_palmata*, *Acropora_cervicornis*, *Favia_favus*, *Halimeda_tuna*, *Lithothamnion_corallioides*, *Hydrolithon_onkodes*, *Lophelia_pertusa*, *Celleporella_hyalina*, *Marginopora_vertebralis*. They represent life-forms (guilds) in what we call the ‘numerical aquarium’.
5. The Knowledge Base concept could become a shared, consensus-written resource.

F. Extracting from the global environmental data:

1. Global data resources that are part of dbSEABED ("http://tinyurl.com/dbseabed") are drawn on to set up a carboLOT work area. Once this is done then the species assemblages, events, etc. can be manipulated locally at the carboLOT folder.
2. The dbSEABED data holdings amount to 12Gb and for this project include GEBCO 1nm gridded bathymetry, World Ocean Atlas 2009, Wavewatch III, MODIS Aqua Chlorophyll and Transmissivity.
3. An option in the "carboLOT_16.py" program does the extraction.