Modeling of the pattern of mangrove resistance to sea-level rise

Denilson da Silva Bezerra
Geuma University (São Luís-MA/Brazil)
Introduction

• In the context of climate change, the sea-level rise is a fact;
• And mangrove ecosystem is very sensitive to this process (For example: mangrove can migrate);
• In addition, intense human occupation can influence the mangrove response to climate change;
• Because the mangrove can not colonize human use areas.
• In this project, we propose a methodological approach, using GIS (Geographic Information System) and modeling based on cellular automata.
Motivation I: The largest continuous mangrove area of the planet is in the Brazilian Amazon (Wilson et al., 2013)
In terms of total area, the Brazil has the second largest mangrove area in the world (Spalding, Kainuma and Collins 2010).
Important! Different authors may have different values. But the fact is ... The Brazil has an extensive mangrove area.
Motivation II: The present study is the first Brazilian modeling experiment; Important: In the context of mangrove response to sea-level rise.
General objective: Simulate the mangrove response pattern to sea-level rise

• **Specific objectives:**
  • Develop a conceptual model;
  • Translate the conceptual model in computer language to allow the simulation exercise;
  • Simulate mangrove response patterns to sea-level rise.
Case Study: Maranhão Island

- Maranhão Island (see figure below) is located on the Brazilian Amazon;
- The Maranhão Island has low relief with altitudes of up to 60 meters;
- Maranhão Island has 146.49 km² mangrove area;
- Tidal rise in the Maranhão Island is 6 m.
Case Study: Maranhão Island
Conceptual Model

- The mangrove responses to sea-level rise depend on factors such as topography (Digital Elevation Model - DEM), tidal range, land cover in adjacent areas, coastal dynamics and mean rate of sea-level rise (Mcleod & Salm 2006, Lovelock & Ellison 2007).

- Our conceptual model have four components: sea-level rise, land cover, environmental restrictions for the mangrove migration (only soil types) and mangrove dynamics (For example, mangrove migration, loss of mangrove area, mangrove resistance to sea-level rise).
Conceptual Model (Figure)
Modeling experiment: BR-Mangrove

• The programming language use: LUA (
  [http://www.lua.org/portugues.html](http://www.lua.org/portugues.html));
• Lua is a powerful programming language, fast and light, designed to extend applications;
• It is a language used in various computer games;
• The modeling platform that was used is the TerraME ([http://www.terrame.org/doku.php](http://www.terrame.org/doku.php)).
TerraME website

TerraME: Simulation and Modelling of Terrestrial Systems

TerraME overview: Modelling Nature-Society Interactions

TerraME is a programming environment for spatial dynamical modelling. It supports cellular automata, agent-based models, and network models running in 2D cell spaces. TerraME provides an interface to TerraLib geographical database, allowing models direct access to geospatial data. Its modelling language has in-built functions that makes it easier to develop multi-scale and multi-paradigm models for environmental applications.
Lua and the Web

Where is Lua?

• Inside Brazil
  – Petrobras, the Brazilian Oil Company
  – Embratel (the main telecommunication company in Brazil)
  – many other companies

• Outside Brazil
  – Lua is used in hundreds of projects, both commercial and academic
  – CGILua still in restricted use

TerraMe Programming Language: Extension of LUA

LUA is the language of choice for computer games

source: the LUA team

[Ierusalimschy et al, 1996]
Geographical Database and Cellular Space

• Our implementation is based on the cellular automata computational model, a logical system which has the concept of cell as the basic unit: each cell has a state and attributes (Wolfram 1983).

• The geographical database and cellular space were organized in the software TerraView (http://www.dpi.inpe.br/terraview/index.php).
the TerraView website
Cells States and Cells Attributes

Figure Cellular space: (A) Empty cellular space (B); Cell states; (C) Attribute-soils; (D) Attribute-Altimetry.
Results

- The simulation considers 88 events of elevation from 0.011 to 0.97 m (period 2012 to 2100);
- The results are presented in terms of total area (ha), mangrove migration (ha) and mangrove inundation (ha).
Results: Br-mangrove in action!
Results: Simulation of sea-level rise
Results: Simulation of mangrove area
Results: mangrove migration and mangrove inundation.
Results: A little movie of the simulation
Final considerations

- The modelling experiment can be used to promote a better understanding of the mangrove responses to potential events of sea-level rise;
- But one must take into consideration environmental and human factors;
- In this context, the human forms of use and occupation in adjacent areas to mangrove must be considered.
And last: the code

```python
# Sea-Level Rise Impacts on Mangrove Ecosystem
# Case Study: Maranhão Island
# Authors: Juliana da Silva Bezerra, Silvana Amaral, Milton Kempel, Pedro Ribeiro de Andrade
# Project funded by CAFES
#
# Cell States and Cell attributes
# SIZE = 0
# MANROVE = 1
# ANTHROPIC_AREA = 3
# ANTHROPIC_AREA2 = 8
# TERRESTRIAL VEGETATION = 4
# MANROVE_SOIL = 5
# MANROVE_SOIL2 = 7
# BEACH = 2
# CHANNEL_RIVER = 2
# MANROVE_MIGRATION = 6
# MANROVE_INFILTRATION = 6
#
# Model Parameters
# Area_cell = 1 -- Cell area in ha
# Initial_time = 1 -- correspondent to 2013
# Final_time = 88 -- correspondent to 2100
# T_elev = 0.011 -- Rate of sea-level rise (m) in a scenario of increase of approximately 0.81 m by 2100 (IPCC, 2013, p.17).
# T_elev = 0.011 -- Tide height on the Maranhão Island (Ferreira, 1989).
# Year Initial = 2013
#
# Database
# ds = CellularSpace(
#   dsType = "RDS",
#   database = "D:\\Banca_mangue_4.2.0\\mangue.db",
#   schema = "Cell_usoa",
#   select = [ "AIL2", "ClassUseo2", "ClassesColor" ]
# )
# os=neighboorhood {
#   strategy = "moore",
#   self = false
# }
# os=synchronize();
#
# Legend
# ClasseLegend = Legend(
#   grouping = "uniquevalue",
#   colorBar = {
#     [value = "SIZE", color = "blue"],
#     [value = "MANROVE", color = "66,111,66"],
#     [value = "BEACH", color = "cyan"],
#     [value = "ANTHROPIC_AREA", color = "yellow"],
#     [value = "ANTHROPIC_AREA2", color = "0,0,0"],
#     [value = "TERRESTRIAL_VEGETATION", color = "31,142,104"],
#     [value = "MANROVE_MIGRATION", color = "0.255,0"],
#     [value = "MANROVE_INFILTRATION", color = "red"
#   }
# )
```
And last: the code

```plaintext
Observers

-- Observer: subject = cs, type = "map", attributes = ("Classless2"), legends = (Classless2Leg)
-- observer: subject = cs, type = "image", DB_HOME = "PATH", "C:\Program Files (x86)\terrain", attributes = ("Classless2"), legends = (Classless2Leg)

-- Node Loop for sea-level rise
for time = Initial_time, Final_time, 1 do
    coord = Coord(x = 68, y = 132)
    cellreference = createCell(coord)
    cellreference.Alz2 = 0.17798

    Mangrove_area = MANGROVE or MANGROVE_migration -- Total area of mangrove in ha
    Mangrove_area_remaining = MANGROVE -- Area of remaining mangrove in ha
    Mangrove_area_impound = MANGROVE_impound -- Mangrove impound in ha
    Mangrove_area_migration = MANGROVE_migration -- Mangrove migration in ha
    SOLI2_AREA_progradation = SOLI2_SOIL2 -- areas of progradation of mud (ha)
    ANTHROPIC_impound = ANTHROPIC_AREA

    --
    forEachCell(cs, function(cell)
        -- Simulation of the rising sea level
        if cell.Classless2 == EE2 and cell.Alz2 > 0 then
            increased_sea = cell.Alz2 + (time * T2_elev)
            Sea_level_rise_in_the_cell_reference
            cellreference.Alz2 = cellreference.Alz2 + (time * T2_elev)
            -- Rate of vertical accretion of mud -- Tmx in mm
            Elev_mn = Elev_cellreference * 1000 -- Sea-level rise in mm
            Tmx = 1.698 + 0.989 * Elev_mn -- Equation proposed by Alongi (2008) with R^2 = 0.704 and p < 0.001
            Tmx = Tmx / 1000 -- Tmx in meter
            -- Increment of the area of tidal influence
            T_w = 1 + Elev_cellreference
            -- Find the lowest neighbor
            countNeigh = 0
            forEachNeighbor(cell, function(cell, neigh)
                if (cell.Alz2 > neigh.Alz2) then
                    countNeigh = countNeigh + 1
            end)
            -- Simulating the advancement of mud banks
            if (cell.Classless2 == MANGROVE_SOIL or cell.Classless2 == CHANNEL_RIVER) and (neigh.Classless2 == MANGROVE_SOIL) then
                forEachNeighbor(cell, function(cell, neigh)
                    if neigh.Alz2 < T_w and
                        neigh.Classless2 == SEE
                    then
                        cell.Alz2 = cell.Alz2 + SOLI2.Area_progradation + Area_cell
                end)
        end)
    end)
```

And last: the code

```python
-- rate of vertical accretion of mud in each cell
if (cell passé Class é sì l e 0 → M A N G R O V E _ S O I L 2 and cell. Class é sì l e 0 → M A N G R O V E _ S O I L 2) or (cell. Class é sì l e 0 → M A N G R O V E _ S O I L 1) and 
   (cell. Class é sì l e 0 = SEE) then
  cell. Alt 2 = cell. Alt 2 + 0.01
end

-- Simulating the flow of water to the neighbors
if (count Neighbors > 0) then
  flux = Increased_see / count Neighbors
end

-- Simulation of the inundation from neighboring
if cell. Class é sì l e 2 = SEE and neigh. Class é sì l e 2 = SEE then
  foreach Neighbors (cell, function (cell, neigh))
    if Increased_see > (neigh. Alt 2 + flux) then
      neigh. Class é sì l e 2 = SEE
    end
  end
end

-- End: Simulation of the sea-level rise

-- Simulation of mangrove migration
foreach Cell (cell, function (cell))
  if (cell. Class é sì l e 0 = M A N G R O V E) then
    foreach Neighbor (cell, function (cell, neigh))
      if (neigh. Class é sì l e 0 = M A N G R O V E) then
        if (2 m > neigh. Alt 2) and 
          (neigh. Class é sì l e 2 = T E S T E R I A L _ V E G E T A T I O N) and 
          (neigh. Class é sì l e 0 = M A N G R O V E _ S O I L 2) then
          neigh. Class é sì l e 2 = M A N G R O V E _ M I G R A T I O N
        end
      end
    end
  end

-- Simulation of mangrove inundation
foreach Cell (cell, function (cell))
  if (cell. past. Class é sì l e 0 = M A N G R O V E or cell. past. Class é sì l e 0 = M A N G R O V E _ M I G R A T I O N) and cell. Class é sì l e 0 = SEE then 
    Mangrove_area_inundation = Mangrove_area_inundation + Area cell
end
```
And last: the code

```python
--- simulation of mangrove migration
forEachCell(0, function(cell)
    if cell.past.ClasseUses2 == TERRESTRIAL_VEGETATION and cell.ClasseUses2 == MANGROVE_MIGRATION then
        Mangrove_area_migration = Mangrove_area_migration + Area_cell
    end
end)

--- simulation of remaining mangrove
forEachCell(0, function(cell)
    if cell.past.ClasseUses2 == MANGROVE and cell.ClasseUses2 == MANGROVE then
        Mangrove_area_remaining = Mangrove_area_remaining + Area_cell
    end
end)

--- simulation of total area (mangrove)
forEachCell(0, function(cell)
    if (cell.ClasseUses2 == MANGROVE or cell.ClasseUses2 == MANGROVE_MIGRATION) then
        Mangrove_area = Mangrove_area + Area_cell
    end
end)

--- simulation of anthropic area inundation
forEachCell(0, function(cell)
    if cell.past.ClasseUses2 == ANTHROPIC_AREA and cell.ClasseUses2 == SEE then
        cell.ClasseUses2 = ANTHROPIC_AREA2
        ANTHROPIC_inundation = ANTHROPIC_inundation + Area_cell
    end
end

Year = Year_initial + time
---
print("Year!", Year)
print("Total area:", Mangrove_area)
print("Remaining area:", Mangrove_area_remaining)
print("Froucation:", SOIL_AREA_progradation)
print("Vertical accretion:", Vert_m)
print("Sea-level rise:", Elev_cellsreference)
print("Tide height!", Z_E)
print("Mangrove migration:", Mangrove_area_migration)
print("Anthropic area inundation:", ANTHROPIC_inundation)
---
if 1 == SS then
    osave(1,"result","ClasseUses")
end
osnotify()
--- Simulation performed with successfully"
```
Thank you very much!

denilson_ca@yahoo.com.br