A simple coupled terrestrial carbon-physics land scheme for use in a closed carbon cycle climate model of intermediate complexity

Mark Williamson
School of Ocean and Earth Science
Southampton Oceanography Centre
Southampton, UK 1

Tim Lenton
School of Environmental Sciences
University of East Anglia
Norwich, UK 2

Ben Adams
Department of Animal and Plant Sciences
University of Sheffield
Sheffield, UK 3

John Shepherd
School of Ocean and Earth Science
Southampton Oceanography Centre
Southampton, UK 4

Andrew Yool
School of Ocean and Earth Science
Southampton Oceanography Centre
Southampton, UK 5

Bob Marsh
James Rennell Division
Southampton Oceanography Centre
Southampton, UK 6

August 24, 2004
1mswl@soc.soton.ac.uk
2t.lenton@uea.ac.uk
3b.j.adams@sheffield.ac.uk
4jgs@soc.soton.ac.uk
5axy@soc.soton.ac.uk
6rma@soc.soton.ac.uk
Abstract

We have developed a simple land scheme for a fully coupled closed carbon cycle climate model of intermediate complexity, GENIE (1-CBM). The land model energy and moisture fluxes are coupled to the terrestrial biogeochemistry and the atmosphere through a number of feedbacks. This report is to act as a user guide and as comprehensive documentation for the simple land scheme (M).

In the simple land model, the land carbon cycle of Lenton (2000) [1] is enhanced by making it spatially explicit and incorporating a number of modifications, some of which include a land hydrological cycle, the effect of water stress on vegetation carbon following Adams (2003) [2] and the 'Leuning' canopy conductance model [6].
## Contents

1 Simple land model components 2
   1.1 Land radiation ........................................... 2
   1.2 Land hydrology ........................................... 3
   1.3 The carbon cycle ........................................... 4
   1.4 Elevation ................................................ 7
   1.5 Albedo .................................................... 7
   1.6 Snow ...................................................... 9
   1.7 Precipitation ............................................. 10
   1.8 Seasonal winds ........................................... 12
   1.9 Emissions ................................................ 12
   1.10 Sea-level change ......................................... 12
   1.11 Off-line testing .......................................... 12

2 Compiling and running the model 16
   2.1 Compiling ................................................ 16
   2.2 The goin file .............................................. 16
   2.3 To run the model .......................................... 17
   2.4 Spinning the model up .................................... 18
   2.5 Continuing a run from a model dump ..................... 18

3 Model outputs 23
   3.1 File names and data format ............................... 23
      3.1.1 Two dimensional arrays ............................... 23
      3.1.2 Land dumps ........................................... 24
      3.1.3 Time series files ..................................... 24
      3.1.4 Global mean air temperature .......................... 24
      3.1.5 Annual average carbon reservoirs and fluxes ......... 25
   3.2 Plotting model output .................................... 25

4 Tuning the model 26

5 Model code structure 28

6 Acknowledgments 29
Chapter 1

Simple land model components

This section will briefly show the equations that represent the various processes built into simple land.

1.1 Land radiation

The land radiation adapts the equations that already exist for the radiation balance between the EMBM (energy-moisture balance model) atmosphere and the ocean (see Edwards and Marsh [3]). The land temperature \( T_l \) evolves according to

\[
h_l c_l \frac{dT_l}{dt} = (1 - \alpha_s)(1 - C_A)Q_{SW} - Q_{LH} - Q_{LW} - Q_{SH}
\]

Where \( c_l = 3.3 \times 10^5 \text{ J/m}^3/\text{K} \) is the specific heat capacity of the land and \( h_l \) is a depth scale. \( \alpha_s \) is the land albedo (see section (1.5)) and \( C_A \) is the absorption of radiation by dust and other particles in the atmosphere. As the land temperature is calculated every ocean timestep (3.65 days at present) the approximation can be made that \( T_l \) reaches equilibrium in this period given that the present radiative forcing does not resolve the diurnal cycle (see Williamson [8] for analytic solutions of the relaxation time for \( T_l \) in diurnal and constant solar forcing cases). If this approximation can be made then all we need do is find the root of equation (1.1) i.e. when it is equal to zero. This is done via a Newton-Raphson iteration scheme which has the advantage of being very quick. It typically takes 2 iterations to find a solution after several ocean timesteps.

The heat flux terms on the right hand side of equation (1.1) are given by the following.

Latent heat

\[
Q_{LH} = \rho_0 L_v E_Q
\]

Where \( \rho_0 \) is the density of water, 1000 kg/m\(^3\) and \( L_v = 2.50 \times 10^6 \text{ J/kg} \) is the latent heat of vapourisation. \( E_Q \) is the total evaporation, a combination of evaporation from bare soil, \( E_l \) and evapotranspiration from vegetated areas, \( E_t \). All have units of m/s.
CHAPTER 1. SIMPLE LAND MODEL COMPONENTS

Net longwave radiation flux between the atmosphere and the land

\[ Q_{LW} = \varepsilon_l \sigma T_l^4 - \varepsilon_a \sigma T_a^4 \]  
(1.3)

Where \( \varepsilon_l = 0.94 \) and \( \varepsilon_a = 0.85 \) are the emissivity of land and atmosphere respectively. \( \sigma = 5.67 \times 10^{-8} \text{ W/m}^2/\text{K}^4 \) is the Stefan-Boltzmann constant and \( T_a \) is the atmospheric temperature.

Sensible heat

\[ Q_{SH} = \rho_a C_H c_{pa} U (T_l - T_a) \]  
(1.4)

Where \( \rho_a = 1.25 \text{ kg/m}^3 \) is a constant surface air density, \( c_{pa} = 1004 \text{ J/kg/K} \) is the specific heat capacity of air and \( U \) is the wind speed in m/s. \( C_H \) is a transfer coefficient for heat and is given by

\[ C_H = \left[ \frac{1}{\kappa} \ln \left( \frac{z_r}{z_0} \right) \right]^{-2} \]  
(1.5)

Where \( \kappa = 0.41 \) is the Von Karman constant, \( z_r \) is a reference height, the same height at which \( U \) is taken, 10 m, and \( z_0 \) is the roughness length of the surface. Presently this is a constant representing ‘average’ global roughness length, \( z_0 = 0.1 \text{ m} \).

All heat fluxes are given in W/m\(^2\) and are completely analogous to those given in C-GOLDSTEIN.

1.2 Land hydrology

Here a new land hydrology scheme is suggested (by TML and MSW) based on Sellers (1992) [4].

Each land grid box has a water reservoir that has a saturated value of \( W^* \) and a present value of \( W \). Water is added to the land by precipitation, \( P_w \) (dimensions m/s) and removed by total evaporation \( E_Q \) and runoff \( R \):

\[ \frac{dW}{dt} = P_w - E_Q - R \]  
(1.6)

Total evaporation \( E_Q \) is a combination of soil evaporation \( E_l \) in the un-vegetated fraction \((1 - f_v)\) of the land surface and evapotranspiration \( E_t \) in the vegetated fraction \((f_v)\):

\[ E_Q = (1 - f_v) E_l + f_v E_t \]  
(1.7)

Bare soil evaporation, \( E_l \) is a conglomerate of both Weaver et al. (2001) [5] and the approach of Sellers (1992) [4] and others. If the grid box is below saturation point / field capacity (i.e. \( W_s < W^* \)) then evaporation has a multiplier \( \beta \), which represents the fact that the land surface is less likely to give up water the drier it becomes. This gives \( E_l \) as

\[ E_l = \beta \frac{\rho_a (q_s(T_l) - q_a)}{r_a} \]  
(1.8)

\( q_s(T_l) \) is the saturation specific humidity for land and \( q_a \) is the atmospheric surface specific humidity (dimensionless units) and both are given in Weaver et al. (2001) [5]. \( r_a \) is the aerodynamic resistance, given by

\[ r_a = \frac{1}{C_W U} \]  
(1.9)

\( C_W \) is a transfer coefficient for momentum and is given by

\[ C_W = \frac{1}{\kappa} \ln \left( \frac{z_r}{z_0} \right) \]  
(1.10)
$C_W$ is the transfer coefficient for moisture and we make the reasonable approximation that $C_W = C_H$. The $\beta$ function is formulated to give a value between 0 and 1 depending on the soil saturation. If $\beta = 1$ then $E_l$ just reverts to evaporation for a surface with freely available water. If $W_s < W_s^*$ then

$$\beta = \left( \frac{W_s}{W_s^*} \right)^4$$

and $\beta = 1$ if $W_s \geq W_s^*$.

Evapotranspiration in the vegetated fraction is calculated by combining aerodynamic ($r_a$) and canopy ($r_c$) resistances to the transfer of water:

$$E_t = \frac{\rho_a}{\rho_0} \frac{(q_s(T_a) - q_a)}{r_a + r_c}$$

Vegetation is assumed to be at atmospheric temperature $T_a$ whereas soil is at $T_l$. Canopy conductance ($g_c = 1/r_c$ in m/s) is based on the model of Leuning (1995) but ignoring the small residual conductance term, $g_0$ (as in Dewar, 1995). It is given by:

$$g_c = \frac{k_3 P}{(pCO_2 - k_{13})(1 + \frac{q_s(T_a) - q_a}{k_{34}})}$$

Where $P$ is photosynthesis in kgC/m$^2$/yr (see next section), $pCO_2$ is the atmospheric concentration of CO$_2$ in ppmv (a surrogate for the leaf surface concentration) and $k_{13} = 29$ ppmv is the compensation point for photosynthesis. The saturation specific humidity of the atmosphere, $q_s(T_a)$, is used as a surrogate for the leaf surface value. $k_{34}$ and $k_3$ are tunable constants. (Note: the values of the $k$ parameters are constantly being refined and so they are not given explicit values in this manual. To find their values look in the file genie-simpleland/data/k_constants.dat.)

Runoff, $R$ (in m/s) can be thought of as the ‘overflow’ of the water buckets. If during a timestep, $\delta t$, $W_s > W_s^*$ then the runoff will be

$$R = \frac{1}{\delta t}(W_s - W_s^*)$$

Where $R$ is always $\geq 0$. This term adds water to an ocean grid box determined by a runoff matrix already present in C-GOLDSTEIN.

Lastly, each grid box’s water reservoir saturation value $W_s^*$ can either be constant (e.g. 0.2 m) or have a linear dependence on soil carbon, $C_s$ i.e.

$$W_s^* = k_9 + k_{10} C_s$$

This represents the difference between the water holding capacity of a desert (sand) and a forest (soil). $W_s^*$ has a maximum value of 0.45 m.

### 1.3 The carbon cycle

The land can store carbon in one of two ways in the model, either in the soil or in vegetation. Each grid box has its own vegetation ($C_υ$) and soil carbon ($C_s$) reservoirs. All the reservoirs have units of kgC/m$^2$ and all fluxes have units of kgC/m$^2$/yr. Vegetation removes carbon from the atmosphere through photosynthesis, $P$, and adds carbon to the atmosphere through plant respiration, $R_υ$. Vegetation also loses carbon to the soil reservoir, through leaf litter, $L$. Finally, $C_s$ loses carbon to the atmosphere by soil respiration, $R_s$. The state of the vegetation is described by both its carbon density $C_υ$ and the fraction of the grid box it covers, $f_υ$. The vegetation carbon reservoir evolves according to

$$\frac{∂C_υ}{∂t} = P - R_υ - L \quad (1.15)$$

The soil reservoir according to

$$\frac{∂C_s}{∂t} = L - R_s \quad (1.16)$$

The carbon flux to the atmosphere from the land is given by

$$\frac{∂C_a}{∂t} = R_s + R_υ - P \quad (1.17)$$

Net photosynthesis, $P$, (i.e. gross photosynthesis minus photo-respiration) is given by:

$$P = k_{18}f_1(CO_2)f_2(W_s)f_3(T_a)f_υ \quad (1.18)$$

This is a product of three functions representing the carbon dioxide fertilisation effect $f_1(CO_2)$, water stress on vegetation $f_2(W_s)$, and air temperature dependance of growth $f_3(T_a)$. The constant $k_{18}$ in kgC/m$^2$ leaf/yr is a baseline leaf photosynthesis rate. $f_υ$ is the vegetated fraction.

If $pCO_2 > k_{13}$ then

$$f_1(CO_2) = \frac{1}{k_{19} pCO_2 - k_{13} + k_{14}}$$

Otherwise $f_1(CO_2) = 0$. $pCO_2$ is the atmospheric concentration of CO$_2$ in ppmv, $k_{13} = 29$ ppmv and $k_{14} = 145$ ppmv (from Lenton, 2000). $k_{19} = (280 - k_{13})/(280 - k_{13} + k_{14}) = 0.6338$ normalises the CO$_2$ response.

If $\frac{1}{2}W^*_s \leq W_s \leq \frac{3}{4}W^*_s$ then

$$f_2(W_s) = \frac{4W_s}{W^*_s} - 2$$

If $W_s < \frac{1}{2}W^*_s$ then $f_2(W_s) = 0$ and if $W_s > \frac{3}{4}W^*_s$ then $f_2(W_s) = 1$.

The temperature response of photosynthesis is based on that of the maximum carboxylation rate of Rubisco, with an additional cut-off term at sub-zero temperatures (following MOSES2/TRIFFID). However, we combine two temperature response functions in order to capture the response of both high and low latitude vegetation types:

$$f_3(T_a) = f_3(T_a) + f_{3a}(T_a)$$
The two functions making up $f_3(T_a)$ are given by

$$f_3(T_a) = \frac{2.0^{0.1(T_a-T_{ref})}}{(1 + e^{0.3(T_a-k_{11})})(1 + e^{-0.3(T_a-k_{12})})}$$

$$f_{3a}(T_a) = \frac{2.0^{0.1(T_a-T_{ref})}}{(1 + e^{0.6(T_a-k_{11a})})(1 + e^{-0.3(T_a-k_{12})})}$$

Where $T_{ref} = 298.15\,K$, $k_{11} = 309.15\,K$, $k_{11a} = 283.15\,K$ and $k_{12} = 268.15\,K$.

The vegetation fraction, $f_v$, is made as a simple saturating function of vegetation carbon. This formulation is likely to change. Vegetation fraction is given by

$$f_v = 1 - e^{-k_{17}C_v}$$  \hspace{1cm} (1.19)

Vegetation respiration depends on the air temperature and the amount of vegetation present (following Lenton, 2000):

$$R_v = \frac{k_{24}}{k_{25}} f_4(T_a) C_v$$  \hspace{1cm} (1.20)

where $k_{24}$ is a vegetation respiration rate (/yr) and $k_{25}$ is a normalising constant.

$$f_4(T_a) = e^{k_{20}}$$

where $k_{20} = 54830\,J/mol$ is an activation energy, and $R = 8.314\,J/mol/K$ is the universal gas constant.

The quantity of carbon lost by the vegetation to the soil via leaf litter is related to the amount of vegetation.

$$L = k_{26} C_v$$  \hspace{1cm} (1.21)

Where $k_{26}$ is in units of /yr, a turnover rate.

Carbon lost to the atmosphere soil respiration is dependent on the land temperature $T_l$ and the amount of carbon in the soil reservoir, $C_s$:

$$R_s = \frac{k_{29}}{k_{30}} f_5(T_l) C_s$$  \hspace{1cm} (1.22)

Where $k_{29}$ is the soil respiration rate (/yr), and $k_{30}$ is a normalising constant.

Above freezing, for $T_l \geq 273.15\,K$:

$$f_5(T_l) = e^{-k_{31}/(T_l-k_{32})}$$

from Lloyd and Taylor (1994), where $k_{31} = 308.36\,K$, $k_{32} = 227.13\,K$. Below freezing, soil respiration rate is assumed to have a constant $Q_{10}$ temperature sensitivity, for $T_l < 273.15\,K$:

$$f_6(T_l) = k_0 Q_{10}^{0.1(T_l-T_0)}$$

where $T_0 = 273.15\,K$, $k_0 = f_5(T_0)$, and $Q_{10} = e^{10k_{31}/(T_0-k_{32})^2} = 4.289$. (This prevents unrealistic blow-up of the soil reservoir as $T_l$ approaches $k_{32}$ in $f_5$.)
1.4 Elevation

The simple land model includes altitude effects. At startup, the model reads in a height above sea-level for each land point and alters the sea-level air temperature calculated in C-GOLDSTEIN according to a lapse rate. The equation for the air temperature over land at height \( z \) (in metres) is given by

\[
T_a(z) = T_a(0) + \Gamma z
\]  

(1.23)

Where \( \Gamma = -6.5 \text{ K/km} \). When this effect is included by setting the `orogswitch` in `goin.simpleland` to 1, altitude effects are also included in the calculation of evaporation, outgoing planetary longwave radiation flux and sensible heat flux (see Weaver et al. (2001) [5] for these equations). Figure 1.1 shows the height of each grid box.

1.5 Albedo

When simple land is compiled together with C-GOLDSTEIN plus any other component, the albedo scheme in C-GOLDSTEIN is replaced in all models by simple land’s own scheme. In C-GOLDSTEIN albedo is prescribed by latitude from a sinusoidal planetary albedo function (a planetary albedo is the resulting albedo from all the other individual albedos of clouds, zenith angle effects and surface colour). This is overwritten by simple land to give a separate surface albedo and a separate atmospheric albedo. The atmospheric albedo \( \alpha_{\text{atm}} \) is made up of two albedos:

\[
\alpha_{\text{atm}} = \alpha_{\text{cloud}} + \alpha_{\text{aero}}
\]

Cloud albedo, \( \alpha_{\text{cloud}} \) is the reflectivity of the global cloud field to shortwave radiation. This is a prescribed monthly field derived from NCEP reanalysis. It
was derived as follows: Cloud radiative forcing, \( C \), is defined as [9]

\[
C = F^{\text{cs}} - F
\]  

(1.24)

Where \( F^{\text{cs}} \) is the clear sky upward radiative flux (W/m\(^2\)) and \( F \) is the long term mean observed upward radiative flux. For solar spectrum radiation this can also be expressed as

\[
C_s = Q_{\text{SW}}(r^{\text{cs}} - r)
\]  

(1.25)

Where \( r^{\text{cs}} \) is the clear sky planetary albedo and \( r \) is the long term mean planetary albedo. In our atmospheric model there are two reflective layers at the top of the atmosphere and at the surface (land or ocean). This allows calculation of \( \alpha_{\text{cl}} \) as

\[
\alpha_{\text{cl}} = \frac{F^{\text{cs}} - F}{Q_{\text{SW}}(1 - \alpha_s)}
\]  

(1.26)

Where \( \alpha_s \) is the surface albedo taken from the prescribed data sets of Matthews [10] and the ocean albedo calculation (see later on in this section).

The other albedo making up \( \alpha_{\text{atm}} \) is the reflection of shortwave radiation due to aerosols (\( \alpha_{\text{aero}} \)). The parameter is a constant, not a field and is tuned to give the correct average global mean temperature, however \( \alpha_{\text{atm}} \) must not exceed 1. \( \alpha_{\text{aero}} \) can be set in \textit{goin.simpleland} and present parameters suggest 0.003 gives a good annual average air temperature.

A parameterisation of cloud albedo was attempted using the available EMBM variables but was unsuccessful. To represent a complex process like this our simple atmosphere does not have enough variables.

The terrestrial surface albedo is dependent on what the surface is i.e. snow, sea-ice, vegetation, bare soil or sand. The terrestrial surface albedo is a function of vegetation and soil carbon. For a snow free gridbox, the terrestrial surface albedo is

\[
\alpha_s = f_v \alpha_v + (1 - f_v) \alpha_{\text{soil}}
\]  

(1.27)

Where \( \alpha_v = 0.1 \), the albedo of vegetation and \( \alpha_{\text{soil}} \) is given by

\[
\alpha_{\text{soil}} = (\alpha_{\text{peat}} - \alpha_{\text{sand}}) \frac{k_{\text{soil}} C_s}{k_{\text{soil}} - k_{\text{sand}}} + \alpha_{\text{sand}}
\]  

(1.28)

Where \( \alpha_{\text{peat}} = 0.11 \) and \( \alpha_{\text{sand}} = 0.3 \). This function has a minimum value of \( \alpha_{\text{peat}} \). If snow is present the terrestrial surface is calculated as

\[
\alpha_s^{\text{snow}} = (\alpha^{\text{snow}} - \alpha_{\text{v}}^{\text{snow}}) e^{-k_{\text{v}} C_v} + \alpha_{\text{v}}^{\text{snow}}
\]  

(1.29)

Where \( \alpha_{\text{v}}^{\text{snow}} = 0.3 \) is the snow covered vegetation albedo and \( \alpha^{\text{snow}} = 0.8 \) is the albedo of a snow covered flat surface.

Calculation of the ocean surface albedo: Ocean albedo is basically Fresnel reflection. Water is a transparent, non-magnetic medium and the reflectivity of the ocean is dependent on the the refractive index contrast across the ocean-atmosphere interface and the incidence angle of the incoming light, the solar zenith angle. The refractive index of sea-water is pretty much constant (\( n \approx 1.339 \)) for the visible wavelength band but the solar zenith angle, \( Z \), varies with latitude, \( \phi \) and the time of year. C-GOLDSTEIN does not resolve the diurnal cycle so the daily average ocean albedo needs to be calculated. It is given by

\[
\alpha_{\text{ocean}} = \frac{\int_0^H R(Z) \cos Z dh}{\int_0^H \cos Z dh}
\]  

(1.30)
CHAPTER 1. SIMPLE LAND MODEL COMPONENTS

Figure 1.2: Annual average planetary albedo

$R(Z)$ is the reflectivity of the ocean based on an empirical formula (Briegleb et al., 1986 [11], but can also be calculated from electromagnetic theory, the Fresnel equations) and $H$ is the angular half-length of the day (see Peixoto and Oort (1991) [12]). The cosine of the solar zenith angle is given by

$$\cos Z = \sin \phi \sin \delta + \cos \phi \cos \delta \cos h$$  \hspace{1cm} (1.31)

Where $\phi$ is the latitude, $\delta$ is the declination of the sun and $h$ is the hour angle from the local meridian where $h = 0$.

This gives $H$ as

$$H = \cos^{-1}(-\tan \phi \tan \delta)$$ \hspace{1cm} (1.32)

$R(Z)$ is given by the following expression:

$$R(Z) = R_{\text{specular}}(Z) + R_{\text{diffusive}}$$

$$R_{\text{specular}}(Z) = \frac{2.6 \times 10^{-2}}{\cos Z + 0.065} + 0.15(\cos Z - 0.1)(\cos Z - 0.5)(\cos Z - 1.0)$$ \hspace{1cm} (1.33)

$R_{\text{specular}}$ is the specular reflection and $R_{\text{diffusive}}$ is the diffusive reflection modelled as a constant ($= 0.06$).

The ocean albedo is calculated via a numerical integration algorithm (an adaptive extended trapezium rule) for each latitude and at each time of the year at the initialization stage of the run and stored in an array.

Figure 1.2 shows the planetary albedo and figure 1.3 shows the surface albedo of each grid box.

1.6 Snow

In simple land there is a simple parameterisation of snow albedo feedback which can be switched on or off using `snowswitch` in `goin.simpleland`. Whether or not
the albedo feedback is switched on the model will calculate fractional snow cover for each land grid box. Snow will be found lying in a land grid box that fulfills the following conditions: $T_a \leq -5^\circ C$ and $T_l \leq -5^\circ C$ and precipitation must be greater than zero. The snow will remain on the grid box until either $T_a$ or $T_l$ are larger than $-5^\circ C$. Figure 1.4 shows the annual average fractional snow cover.

1.7 Precipitation

In goin.simpleland there is an option to turn on or off an alternative precipitation parameterisation than that used by C-GOLDSTEIN. In C-GOLDSTEIN precipitation falls in a grid box if the relative humidity of the air exceeds $r_{max}$, a value you may specify in goin.simpleland. This is usually set to 0.85 (a relative humidity of 85%). The amount of precipitation falling is the amount of moisture needed to be removed from the air to make the relative humidity $r_{max}$ again. In simple land, our alternative function is based on the same principle but instead of instantaneous precipitation, we relax the value of the relative humidity towards $r_{max}$ with a timescale $timepptn$ (specified again in goin.simpleland), usually 5 days. The modified equation for precipitation is given in equation (1.34).

$$P = \frac{1}{T} \frac{\rho_a h_a}{\rho_o \delta t} (q_a - r_{max} q_s(T_a))$$ (1.34)

Where $h_a$ is the height of the atmospheric boundary layer for moisture, $T$ is $timepptn$ and $\delta t$ is the length of the time step. This allows moisture to reach further into drier areas of the globe. Figure 1.5 shows the annual average precipitation with this scheme.
Figure 1.4: Annual average fractional snow cover

Figure 1.5: Annual average precipitation in m/s
1.8 Seasonal winds

Also included in the simple land model are seasonal prescribed wind vector fields for advecting moisture and temperature in the atmosphere and seasonal prescribed wind speeds for calculating sensible heat fluxes and evaporation. Seasonal winds change moisture movement at different times of the year allowing the model to simulate monsoons etc. more effectively.

1.9 Emissions

Simple land also includes an option for forcing the model with carbon emissions from a time series file located at `genie-simpleland/data/emissions_timeseries.dat`. When the emissions forcing option is changed to ‘y’ in `goin.simpleland` the model adds carbon to the atmosphere. The first column in the `emissions_timeseries.dat` is the time in years from the beginning of the run. The second column is carbon emissions in GtC/yr (giga-tonnes of carbon per year). The model linearly interpolates between the values in the timeseries file. This option maybe useful to select if you wish to do global warming experiments once you have spun the model up into a steady state.

1.10 Sea-level change

A simple calculation is performed by simple land to get the change in sea-level height from a reference number due to thermal expansion. The reference number used in the calculation is that of global average sea-water density that can be specified in `goin.simpleland`. The variable name is `rhoref`. The model then calculates what the change in sea-level height is according to the present global average sea-water density. The calculation assumes that the ocean basins have vertical sides so any expansion by the water is upwards and cannot spread onto or away from the continents. To use this option effectively, run the model to a steady state, use the final value of the global average ocean density printed to a timeseries file in the model’s output directory (filename `sealevel`) and run the model through an emissions scenario.

1.11 Off-line testing

Simple land may also be run ‘off-line’. When this option is selected in `goin.simpleland` simpleland no longer sees the coupled model’s variables, it sees monthly data sets taken from NCEP i.e. the model is forced with observed data. The observed fields it is forced with are precipitation, relative humidity and air temperature. The shortwave radiation flux entering the land still comes from the coupled model however. This is a useful tuning and diagnostic tool but while running the model in offline mode water is not conserved globally. The ocean and atmosphere model still see precipitation as calculated by the EMBM but evaporation and runoff for the land are calculated from the NCEP data. This means water is not conserved globally and generally the ocean gets progressively fresher. This means ocean and atmosphere states when putting simple land in offline mode
should not be used. However the simple land restart file is useful for getting realistic vegetation in continuing runs.

Figures of the vegetation and soil carbon in off-line mode are shown in figure 1.6 and figure 1.8 respectively. Observations of these two quantities are shown in figures 1.7 and 1.9.
CHAPTER 1. SIMPLE LAND MODEL COMPONENTS

Figure 1.7: Annual average vegetation carbon from observations (kg C/m$^2$)

Figure 1.8: Annual average soil carbon in off-line mode (kg C/m$^2$)
Figure 1.9: Annual average soil carbon from observations (kg C/m$^2$)
Chapter 2

Compiling and running the model

This section describes how to compile a version of the model that includes C-GOLDSTEIN (ocean and EMBM atmosphere), BioGeM (ocean biogeochemistry), AtChem (well mixed box atmospheric chemistry) and simple land to create a closed carbon cycle model, the executable CBM-GOLDSTEIN.

2.1 Compiling

To compile the model, go to the directory genie-cgoldstein and type the command

`make cbm_goldstein`

This will compile all the subroutines that make up the complete model. You may have some problems with compiler flags depending on your machine’s architecture and installed compilers. The flags for the various machines are contained in the file Makefile in the same directory. If you have problems compiling then comment out the present ‘FLAGS = -r8 -o’ with a ‘#’ and try another below it. ‘FLAGS = -r8 -o’ works with SGI machines and ‘FLAGS = -xtynamap=real:64 -o’ works with Sun workstations. After successful compilation you should have an executable called `cbm_goldstein` sitting in the same directory.

2.2 The goin file

There is a goin file for C-GOLDSTEIN with BioGeM called `goin_biogem` which lives in the directory genie-cgoldstein and a separate goin file for simple land in the directory genie-simpleland. `goin_biogem` sets the length of the run, how often to write data, and the values of C-GOLDSTEIN’s ocean and atmosphere parameters.

The key parameters you may want to alter are the length of the run, `nsteps`, in ocean time steps (the number of ocean time steps per year is given by the number the second row down, usually 100). `npstp` controls how often the model writes information to a goout file (in ocean time steps). `iwstp` tells the model how often to write out a full dump, i.e. the current state of the model that can
be used for a restart from the same point. \texttt{itstp} tells the model how often to write to the time series files and \texttt{ianav} tells the model how often to average the model’s state over i.e. a setting of 100 with the number of ocean simple land per year also set to 100 will produce a file of the model’s average state over a year, the annual average.

In the second row you can chose whether to start from a restart file (put option ‘c’) or to start the model from a homogenous state and ‘spinup’ the model (put option ‘n’). If you chose option ‘c’ then you must type the name of the restart file in the \texttt{input filename} row. The \texttt{output filename} row specifies where you want the model output to be written to.

For a proper explanation of this file please see the BioGeM and/or C-GOLDSTEIN documentation.

The second goin file, \texttt{goin.simpleland} is hardwired into the simple land code so you must not change its name. In this file you can specify how often simple land is called in ocean timesteps, \texttt{msimpleland} (usually 5), the initial sizes of the global carbon reservoirs in GtC (giga-tonnes of carbon) for vegetation (\texttt{Cvegini}) and soil (\texttt{Csoilini}). Note that \texttt{Catmini} does not specify the size of the atmospheric carbon reservoir in CBM-GOLDSTEIN. In this executable, the parameter is redundant. \texttt{fvini} specifies the initial vegetation fractional cover in each land box.

\texttt{rmax} and \texttt{timepptn} control the precipitation scheme. \texttt{rmax} is the relative humidity (fractional) and \texttt{timepptn} is the time scale in days. \texttt{aeroalb} is the albedo due to aerosols and \texttt{rhoref} is the reference global ocean density which is used to calculate change in sea-level height.

Next in the file follow a set of boolean logic switches. \texttt{carbonswitch} turns on/off carbon feedbacks to the climate. These feedbacks only include surface albedo, transpiration feedbacks and soil carbon feedbacks on the water holding capacity of the land. If this is set to off then evaporation is calculated for bare soil only and surface albedo is constant. \texttt{orogswitch} turns the altitude effects on/off and \texttt{pptnswitch} does the same for the precipitation scheme. \texttt{snowswitch} turns the snow albedo effect on/off. If \texttt{offlineswitch} is set to 1 then the model reads in and is forced with NCEP data (to be used for diagnostic and tuning purposes only).

\texttt{iniq} controls the initial size of the water buckets $W^*_s$ (in metres). The next line is whether or not to force the model with carbon emissions from the time series file \texttt{emissions_timeseries.dat}. Select ‘y’ or ‘n’. The last two rows control whether you want to start the model from a previous model dump and if yes, the file’s name and location.

\subsection{To run the model}

Make sure you’re in the directory \texttt{genie-cgoldstein} and have produced the executable \texttt{cbm_goldstein}. You also need to create a directory \texttt{results} on the same level as the directory \texttt{genie-cgoldstein}. After setting the desired parameters in \texttt{goin.biogem} and \texttt{goin.simpleland} you are ready to run the model. To run the model type

\begin{verbatim}
  cbm_goldstein < goin.biogem
\end{verbatim}

This should set the model running and printing information to the screen. To print this information to a file instead of the screen type
2.4 Spinning the model up

Producing a model state that looks like the pre-industrial world is the usual starting point for transient experiments such as global warming. Start the model from a homogenous state (the ‘n’ option), select the radiative CO$_2$ feedback to be turned on and run for 5000 years to obtain a steady state. Make sure you’re restoring the atmospheric CO$_2$ concentration to 278 ppm to get the ocean and atmospheric carbon inventories roughly correct. This is one of biogem’s options and can be set in `gem.config.atm.par`. Once the model has reached a steady state you may like to check everything is in equilibrium by turning the carbon restoring off in biogem and running for a further period of time. This should produce the pre-industrial steady state as a starter for transient runs. The resulting vegetation state is shown in figure 2.4. Note that the Sahara is green as is the Australian desert. These regions are bistable in the model. To obtain a starting vegetation state that looks more like the present earth you need to take a few more steps in the spin up process. To get a vegetation state as shown in figure 2.5 you need to restart the model with an offline simple land restart file but while using the biogem, atchem and c-goldstein restart files that you obtained from the green Sahara vegetation state. This is easily done as you can specify a different simple land restart file name in `goin.simpleland` to the one specified in `goin.biogem`. You will need to spin the model for a further period of time once you’ve substituted the restart file to allow for everything to get back to a steady state. The boreal forest will die back and a hole will open in the Amazon vegetation due to deficiencies in the EMBM but the desert regions will not re-green.

Some figures of this final model state are shown. Figure 2.1 shows the annual average air temperature. Figure 2.2 shows the annual average land temperature.

2.5 Continuing a run from a model dump

After you’ve finished spinning the model up and you’re happy that it’s in a steady state you can use the model’s output files as a starting point for transient experiments. If you specified the model output to have the name `ini`, then the model dump files are:

- `ini.1` - C-GOLDSTEIN file
- `ini.1.biogem` - BioGeM file
- `ini.1.atchem` - AtChem file
- `ini.sland.1` - Simpleland file

All these files need to be in a directory named `results` which is the same directory that the model will send it’s output to. It’s important to note that when specifying input and output names in `goin.biogem` that they are no more or no less than 7 characters long for output file names and exactly 9 characters long for input file names. To continue a run using these files just change the ‘n’ to a ‘c’ in the second row of `goin.biogem`, type a name to call the output files and then finally change the last two lines of `goin.simpleland`. All that remains is
Figure 2.1: Annual average air temperature (°C)

Figure 2.2: Annual average land temperature (°C)
CHAPTER 2. COMPILING AND RUNNING THE MODEL

Figure 2.3: Annual average water bucket height (m)

Figure 2.4: Annual average vegetation carbon (kg C/m²) after stage 1 of spin up. Note the green desert regions.
CHAPTER 2. COMPILING AND RUNNING THE MODEL

Figure 2.5: Annual average vegetation carbon (kg C/m²) after stage 2 of spin up. Note no green desert regions.

Figure 2.6: Annual average soil carbon (kg C/m²) after stage 2 of spin up.
to set how long you wish to run the model for and the various other parameters in both goin files.
Chapter 3

Model outputs

The model produces a number of output files in the directory \textit{results}. In this section the names of the files and the format in which they produce the data will be given for the simple land component only. For C-GOLDSTEIN and BioGeM outputs please see those respective user manuals.

3.1 File names and data format

3.1.1 Two dimensional arrays

The simple land model produces a number of files that are output as one long column of numbers of length $36 \times 36$. These files are of one variable and have one value for each point of the C-GOLDSTEIN grid (i.e. the grid is $36 \times 36$). Listed below are these files with their units:

- \texttt{filename.lqavg} - The amount of water in each grid box averaged over the last \texttt{ianav} ocean time steps (units meters).
- \texttt{filename.itavg} - The temperature of the land in each grid box averaged over the last \texttt{ianav} ocean time steps (units $^\circ$C).
- \texttt{filename.palbavg} - The effective planetary albedo for each grid box averaged over the last \texttt{ianav} ocean time steps.
- \texttt{filename.snowavg} - The fractional snow cover for each grid box averaged over the last \texttt{ianav} ocean time steps.
- \texttt{filename.albsavg} - The surface albedo for each grid box averaged over the last \texttt{ianav} ocean time steps.
- \texttt{filename.runavg} - The runoff for each grid box averaged over the last \texttt{ianav} ocean time steps (units m/s).
- \texttt{filename.pptnavg} - The precipitation for each grid box averaged over the last \texttt{ianav} ocean time steps (units m/s).
- \texttt{filename.relhavg} - The relative humidity for each grid box averaged over the last \texttt{ianav} ocean time steps.
- \texttt{filename.rcslavg} - The canopy resistance for each grid box averaged over the last \texttt{ianav} ocean time steps (units s/m).
CHAPTER 3. MODEL OUTPUTS

3.1.2 Land dumps

These are files that are produced to allow a perfect restart of the model. They contain all the information the model needs to know to calculate any other variable it needs, in other words this file contains all the information about the model’s state. There are two files produced, both in the same format. These are filename.sland.\(n\), where \(n\) is an integer from 0 to 9 and filename.sland.avg. Only filename.sland.\(n\) allows perfect restarts as this is a ‘snapshot’ of the model’s state at a particular time. filename.sland.avg is the complete model’s state but averaged over the last \(i\)anav ocean timesteps.

The data is printed in the following order (all in one column):
- photosynthesis: 1 to size of grid (should be \(36 \times 36 = 1296\)) in units of mol C /gridbox/yr.
- vegetation respiration: 1 to size of grid in units of mol C /gridbox/yr.
- leaf litter: 1 to size of grid in units of mol C /gridbox/yr.
- soil respiration: 1 to size of grid in units of mol C /gridbox/yr.
- vegetation carbon: 1 to size of grid in units of mol C /gridbox.
- soil carbon: 1 to size of grid in units of mol C /gridbox.
- land temperature: 1 to size of grid in units of °C.
- land water: 1 to size of grid in units of meters.
- snow fraction: 1 to size of grid, no units.

3.1.3 Time series files

Simpleland produces a timeseries file of globally averaged quantities. This file is written to every \(i\)tstp ocean timesteps and is called filename.slandt. The data is formatted as follows:
- column 1: vegetation carbon (GtC)
- column 2: soil carbon (GtC)
- column 3: atmospheric carbon (GtC)
- column 4: vegetation fraction
- column 5: CO\(_2\) concentration (ppm)
- column 6: photosynthesis (GtC/yr)
- column 7: vegetation respiration (GtC/yr)
- column 8: leaf litter (GtC/yr)
- column 9: soil respiration (GtC/yr)

Simple land also produces a timeseries file of average global ocean density and sea-level height relative to the reference density rhoref. It is written to every \(i\)tstp ocean timesteps and is called filename.sealevel. The data is formatted as follows:
- column 1: time (units in C-GOLDSTEIN non-dimensional time).
- column 2: change in sea-level height relative to rhoref (m).
- column 3: global average ocean density (kg/m\(^3\)).

3.1.4 Global mean air temperature

This is actually calculated in c-goldstein but is added as one of simpleland’s outputs for convienience. The file is filename.gmairt. The file is a timeseries but is written to every \(i\)anav ocean timesteps rather than the usual \(i\)tstp. It has the following format:
column 1: time (in years from the start of the present run) 
column 2: global average air temperature (°C).

3.1.5 Annual average carbon reservoirs and fluxes

This is a timeseries file that calculates global carbon reservoir sizes and fluxes averaged over the last \textit{ianav} ocean timesteps and written to at this frequency. For this file to be meaningful you should select \textit{ianav} to be the same as \textit{nyear}, the length of a year for global annual average fluxes. This file is also very useful for tuning and finding out what’s going on in the global carbon cycle. It has the following format:

- column 1: year from the beginning of the run
- column 2: global annual average net photosynthesis (GtC/yr)
- column 3: global annual average vegetation respiration (GtC/yr)
- column 4: global annual average leaf litter (GtC/yr)
- column 5: global annual average soil respiration (GtC/yr)
- column 6: global annual average vegetation carbon (GtC)
- column 7: global annual average soil carbon (GtC)
- column 8: global annual average vegetation fractional cover
- column 9: global annual average atmospheric carbon (GtC)
- column 10: global annual average ocean carbon (GtC) (only for cbm\textsubscript{goldstein})

3.2 Plotting model output

In the directory \texttt{genie-simpleland/plot} you will find some matlab scripts that can be used to plot simple land output. \texttt{simpleland\_constants.m} and \texttt{simpleland\_mask.mat} are not plotting scripts but are called by the other scripts as they contain the common variables. \texttt{simpleland\_twoDarray.m} will plot the two dimensional arrays. By typing \texttt{simpleland\_twoDarray} in matlab you will be asked to enter the run id and then be presented with the option of plotting either water bucket fullness, land temperature, run off, fractional snow cover or planetary albedo. \texttt{simpleland\_carbon.m} lets you plot carbon flux fields i.e. photosynthesis, soil respiration or carbon reservoirs and vegetated fraction. \texttt{simpleland\_timeseries.m} produces an array of plots from \texttt{slandt}. \texttt{simpleland\_avgt.m} will let you plot numbers and derived quantities from \texttt{slavgt} including global annual average land carbon fluxes, carbon inventories, net ecosystem productivity (NEP) and net primary production (NPP). There is also another script, \texttt{simpleland\_evap.m} that allows you to plot evaporation, transpiration and evapotranspiration fields from the file \texttt{evapavg}.

You need to specify the location of your model output in \texttt{simpleland\_constants.m}. Where ‘path’ is specified in this script put your own file location.
Chapter 4

Tuning the model

In this chapter we’ll only discuss tuning the simple land model and not the other components of cbm_goldstein. Simple land as it comes should produce pre-industrial land carbon fluxes and land carbon inventories when in offline mode. These numbers obtained from the 1995 IPCC report are 550 GtC in vegetation carbon, 1500 GtC in soil carbon, 100 GtC/yr in net photosynthesis and 50 GtC/yr for vegetation respiration, leaf litter and soil respiration. Tuning in offline mode is made very easy as many of the feedbacks between the carbon model and the climate are fixed. This makes the effect of changing the carbon constants stored in `k_constants.dat` almost linear in response. With just one run you can calculate the values of the $k$ constants that give you your desired fluxes i.e. the terrestrial carbon flux equations reduce to

\[ P = k_{18} K_{env} f_u \]

\[ R_u = \frac{k_{24}}{k_{25}} K_{env} C_u \]

\[ L = k_{26} C_u \]

\[ R_s = \frac{k_{29}}{k_{30}} K_{env} C_s \]  \hspace{1cm} (4.1)

Where the fluxes are global annual averages (GtC/yr) and $C_v$ and $C_s$ are the global annual average reservoir sizes (GtC) read from the file `.slavg`. After one run and a rearrangement of the above equations to determine the values of the environmental constants (the $K_{env}$) you can tune to any fluxes and any reservoir sizes.

One other thing to note is that if you change $k_{18}$, the photosynthesis rate then you must also adjust the canopy conductance model constants $k_3$ and $k_{34}$ as canopy conductance depends on photosynthesis. We tune this constants to give the same response as Cox et al. (1998) as shown on their figure 6 [13]. Because their photosynthesis function and canopy conductance model is different you cannot reproduce these figures exactly. In the plot directory in the genie-simpleland code there is a matlab script that lets you tune the canopy conductance.
Figure 4.1: Desired response for canopy conductance. Figure obtained from MatLab script `canopy_tune.m`

Another tuning target is the global annual mean pre-industrial air temperature. Today’s global annual mean air temperature is 15 °C. Assuming a CO$_2$ concentration of 350 ppmv for today and 280 ppmv for the pre-industrial state, a quick calculation reveals that you should tune to an air temperature of 14.5 °C. This can be done by altering `aero_alb` in `goin.simpleland`. For our usual goldstein parameter settings this turns out to be around 0.003.
Chapter 5

Model code structure

Simpleland is structured much like C-GOLDSTEIN in terms of file names. Before the main time loop simple land is initialised by gsetlsimple.f. This subroutine reads from goin.simpleland and sets up the initial state of the model. If the run is a continuing run then gsetlsimple.f calls another subroutine inmlsimple.f which reads in the model dump file. During the initialisation stage, two other subroutines are called: setup_emissions.f is called from mains.F and creates an array of carbon fluxes to feed the atmosphere with throughout the run from the emissions time series emissions_timeseries.dat. The other is ocean_alb.f which is called from radfor.F that calculates the ocean albedo at each latitude and istep before the main time loop.

During the main time loop in mains.F there are calls to two different subroutines. One is a call to surflux.F, actually part of C-GOLDSTEIN, that calculates heat and moisture fluxes and updates these values. This subroutine is called every ocean timestep. The second call is to carbon.F which calculate carbon fluxes and updates reservoirs. This subroutine is called every msimpleland ocean time steps. These two subroutines form the heart of simple land.

There are two diagnostic subroutines, diaglsimpleosc.F which calculates the filename.sland.avg files and diagsimplet.f which calculates the time series file filename.slandt. These are called every ianav and itstp ocean time steps respectively.

The last subroutine outmlsimple.f is called every iwstp and outputs the full model dumps filename.sland.n.

The remaining files are data sets and are found in the directory data. k_constants.dat is a list of constants used in the carbon model. orography.dat are the elevations above sea-level. monthly_uwind.silo, monthly_vwind.silo and monthly_windspeed.silo are the u and v components of the wind for each month and the wind speed for each month respectively. cloud_albedo_monthly.dat contains the mean monthly cloud albedo fields and all files prefixed with ‘NCEP’ are the files used to force the model in off-line mode.

Lastly var.simpleland.cmn contains all the common blocks of the model.
Chapter 6

Acknowledgments

Mark Williamson, Tim Lenton and John Shepherd were supported by the Tyndall Centre for climate change research. Andrew Yool was supported by the GENIE project and Bob Marsh and Ben Adams were supported by NERC funding.

Of the other components that make up cbm-goldstein, Andy Ridgwell (BioGeM) was supported by the Tyndall Centre and Neil Edwards is supported by ?.
Bibliography


