

G-Range: Development and Use of a Beta Global Rangeland Model

Randall B. Boone, Richard T. Conant, and Thomas E. Hilinski

Natural Resource Ecology Laboratory, 1499 Campus Delivery, Colorado State University, Fort Collins, Colorado, USA 80523-1499.

Contact R.B. Boone at: Randall.Boone@ColoState.edu

August 31, 2011

Preface

In April of 2010, Drs. Mario Herrero and Philip Thornton of the International Livestock Research Institute contracted with Drs. Boone and Conant to create a global rangeland model of moderate complexity. Boone was funded for a 50 day effort, and Conant for ca. 40 days. An opportunity to prepare a manuscript for a special issue of the *Proceedings of the National Academy of Science* arose, and Conant took the lead in that effort. Boone created the rangeland model, called G-Range, with input from Conant, drawing upon existing models and new information (see Acknowledgements). Conant and Hilinski reviewed the logic used in G-Range to simulate ecosystem processes, and joined to create the initial baseline simulation and conduct the scenario reported in this report. This report summarizes that effort, and provides guidance on using the accompanying simulation software. The structure of this report was influenced by the ODD (Overview, Design Concepts, and Details) protocol described in Grimm et al. (2006; 2010).

Overview

Purpose

Temperate and tropical rangelands occur throughout the world and are the most extensive land type on earth. These lands are biologically diverse, and support the livelihoods of millions of households. In most rangelands, there is not sufficient precipitation for agriculture, and so families use livestock to essentially turn sunlight into food (Hobbs et al. 2008). Historically, the inhabitants of these areas had to contend with droughts, fires, livestock raids, and other stressors and shocks. Great variability in precipitation and the climatic extremes that characterize semi-arid and arid areas led to coupled natural and human systems that were inherently flexible; the adaptive capacity of the system components – their ability to adapt to a changing environment – was high. Ungulates evolved physical and behavioral adaptations such as high heat tolerance and migration, and people adopted behaviors such as transhumant movement and developed cultural norms that minimized exposure to stresses. Flexibility came to be a defining feature of pastoral communities, and their adaptive capacity was great (Figure 1, ‘Historical’).

During the 20th century, adaptive capacity in semi-arid and arid systems began to erode. Changes in land use and fragmentation limited movements by wild and domestic ungulates and reduced the forage they could acquire, human-wildlife conflicts increased, and wildlife consumption expanded. Land use intensified through expanded agriculture, increased livestock stocking, and changes in water supplies. Land tenure changes, such as colonial domination, subdivision, the creation of parks, and privatization reduced transhumance. Government policies forced changes that induced sedentarization, which reduced options available to pastoralists to adapt to environmental change. Human population growth severely limited the capacity of pastoralists to adapt, and some pastoralists themselves reduced their adaptive capacity, for example, by settling permanently to be near schools and services. Whether extrinsic to systems (e.g., government policies) or intrinsic (e.g., people wishing to be near services), the adaptive capacity of semi-arid systems has been reduced (Figure 1, 'Recent').

The 21st century will bring unprecedented climate change to semi-arid and arid systems (at least within human history). Our use of fossil fuels is releasing the carbon long-stored in oil and gas into the atmosphere. Higher carbon concentration (and concentrations of other gases) is increasing the earth's greenhouse effect. In general, scientists have projected that in the 21st century temperatures will increase, precipitation will increase in some places and decrease in others, the variation in precipitation will increase between years, and the frequency of climatic shocks such as drought will increase. These stressors will reduce further the adaptive capacity of coupled semi-arid and arid systems (Figure 1, 'Future').

One means of increasing the likelihood that households, enterprises, and land managers will be able to adapt to future changes is if future conditions can be anticipated. A quantitative method of anticipating future conditions is through using simulations, where inputs into simulations are modified as predicted in change scenarios (Peck 2004). We sought a means of simulating future changes in rangelands across the globe. The simulation needed to be responsive to changes in precipitation and temperature through time. We developed a process-based simulation model that is spatially explicit and of moderate complexity. The model allows global analyses of native rangelands and the changes they may undergo. The tool we developed is a global rangeland model called G-Range.

G-Range Design Goals and Concepts

Several goals and constraints guided the design of G-Range. These include that we sought:

- A simulation tool for global rangelands that captures main primary production, and its dynamics;
- A tool of moderate complexity, one that could be useful to a new user in a week or less;
- A structure that includes simulating changes in all rangelands across the globe within a single executed process;
- A monthly time-step in simulations;

- Representation of global vegetation at least at the scale of herbaceous, shrubs, and trees;
- The ability for the proportions of those different kinds of plant types to change over time;
- Simulations that may span from about 5 to 100 or more years;
- The ability to include natural or management modifications to rangelands, such as through fire or fertilization;
- Programming structures that will allow the software to be parallelized for use on multi-processor clusters or networks, although making the software run parallel was not part of this effort;
- Greater concern for clarity in logic and ease of use than in conserving hard drive space or algorithmic elegance;
- Portability in the G-Range code, allowing simulations to be done on a variety of platforms (e.g., Windows, Linux cluster);
- A graphical user interface (GUI) that is weakly linked to G-Range. G-Range should be able to be run in batch model, without input from a user, and related to that;
- Parameter files used, rather than inputs from a GUI, allowing simulations to be made without using the interface;
- Output should be straightforward spatial surfaces, without complex summary analyses. Those post-simulation analyses may more readily be done in other packages based on output from G-Range;

G-Range was not intended to be programmed ‘from scratch.’ Completing such an effort would take more time than we had available to dedicate to the project, and would duplicate the efforts of many researchers. Numerous grassland and rangeland simulation models have been developed. We explored a variety of models, to different degrees (i.e., SimSAGS, MAPSS, IBIS, the Hurley Pasture Model, GEM-1, Biome-BGC, GENDEC, Grazing Lands Application, GRAZPLAN, PHYGROW, and Pasture Quality Model, SAVANNA, and Century). Some models are quite complex, making global application unwieldy. Other models are too simple, not allowing for scenario analyses of the types we intend. Some simulation models are point-based, inappropriate for a spatially explicit global simulation model. Some models use simple rules to infer changes, which can be unrealistic. For example, logic in a model may dictate that a drought of a given severity causes a given percentage of herbaceous plants to die. These models are inappropriate for our use. More practically, some were judged out-of-date based on their Web sites. Others appeared ‘closed source’ rather than ‘open source’ packages, such that the suitability of the software for our use would be difficult to judge without requests to the authors. Some were commercial products and were excluded from consideration.

We selected Century to serve as the foundation for the soil modeling and physiological aspects of the G-Range model. There were both scientific and practical reasons for this. Some of the main uses of G-Range are related to climate change, and the potential for rangelands to sequester carbon. Century is the model most commonly used to explore questions about soil organic

carbon in agroecosystems around the world, and is used very often in rangelands as well. Century has been in development for more than 20 years, and has been assessed by comparing its results to observed data many times (e.g., Century biography at NREL 2000). We therefore judged the model suitable to represent carbon relationships and plant growth. More practically, Century was begun and continues to be developed at our home laboratory, the Natural Resource Ecology Laboratory at Colorado State University. Indeed, our offices neighbor those of main authors of Century, Drs. William Parton and Dennis Ojima (see Acknowledgements). Lastly, the authors of Century were supporting of our efforts to use it as a foundation for G-Range. There remains the possibility that one of the other models listed above make it more attractive than Century as a basis for G-Range. Regardless, we were much more likely to be able to ask questions and interpret the logic in the Century model correctly given the large knowledge base at our laboratory. Aspects of G-Range were influenced by SAVANNA as well, which is authored by Dr. Michael Coughenour (see SAVANNA biography at NREL 2007), also of our laboratory. Boone has used SAVANNA in research for more than a decade, and so its structure influenced the design of G-Range. Lastly, plant population modeling and some other aspects of G-Range are new contributions.

Representing Ecosystem Processes across Space

Century is a point-based model. One may conceptualize a simulation with Century as occurring on a representative square meter plot within the homogeneous landscape of interest, such as a forest patch, agricultural plot, or US county. This is not a specific square meter identified in the simulation, simply a helpful conceptualization. To do simulations, analysts using Century make many simulations to represent a study area, one for each area that is considered homogeneous in its ecosystem processes. For example, simulating the counties in the US may entail summarizing results from 3000+ simulations, one for each county, using tools to manage the related simulations (see NREL 2010).

This rationale is used in G-Range as well, although the software is designed to incorporate the simulations required into a single executable process. For every rangeland landscape cell simulated, one may conceptualize there being a representative square meter plot for which ecosystem dynamics are being simulated. In G-Range, for every cell dynamics are simulated for this representative square meter, such as soil, water, carbon, nitrogen, and decomposition. (Century tracks phosphorous and sulfur as well, but G-Range, like SAVANNA, does not). Also simulated for the representative square meter (as in Century) are the growth and death of plant parts.

As a point-based simulation, Century does not need to represent plant population dynamics. In contrast, G-Range populates global rangelands with plants, and we sought to have the types of plants (herbs, shrubs, trees) to be able to change in proportion over time. We therefore incorporated a means of representing plant populations into G-Range. This was done by

incorporating for each landscape cell a representative square kilometer. Again, these square kilometers were not specified in the model, but are abstractions included for each cell; indeed, a cell may be smaller than a square kilometer and the rationale still holds. Plants have a crown dimension assigned in G-Range and crowns do not overlap (or root dimension if larger and not overlapping, the logic remains the same). Each plant in the population occupies an area the size of its crown dimension squared. The proportion cover of a given plant type is therefore equal to the proportion of the square kilometer patch the plant type occupies, and the proportion bare ground is the square kilometer minus the sum of the three plant types. That said, plants may occur in the understory of trees, for example, and those are tallied as a separate population (see below). Whole plant death rates may then be influenced by a nominal rate, water availability, grazing, shading, and seasonality for annuals. Whole plant establishment may be influenced by relative seed production, water availability, crowding from surface litter or herbaceous roots, and by shading from woody cover. Together, these tensions for plant death and plant reproduction yield changing plant populations that are sensitive to changes in the ecosystem, such as precipitation rates.

Note that the plant production portion of G-Range and the plant population portion are not tightly linked. Attributes are shared when needed between the subroutines as needed. But in general, the square meter simulated for a cell is representing an area with vegetation. That conceptualized patch is going to include some portion of herbs, shrubs and trees. The plant population model may simulate that the cell is dominated by bare ground. An addition that may be made to G-Range (see Future Steps) is a set of products that integrate these two views. For example, a desert with moderately productive plant that had very low cover would show average net primary productivity across the landscape as quite low.

Entities, State Variables, and Scales

The entities that comprise a G-Range simulation are landscape cells. Global spatial layers (i.e., digital maps) are composed of a matrix of cells. Each cell represents a whole or fraction of a degree of earth's surface. A portion of those cells represent land, and a portion of those are classified as rangelands. Users may define the cells considered rangelands. Ecosystem processes are simulated for these rangeland cells.

Numerous parameters and state variables are used in G-Range. The variables are a subset of those used in the Century that were judged to be most critical to include in a rangeland simulation of moderate complexity, and that have varied in past applications of Century and SAVANNA (some of SAVANNA was adapted from Century, and so there is overlap). Some parameters that are the same across past applications were '*hardwire*' coded into G-Range to simplify applying the model (depending upon the context, italics are used for entries in the Glossary and to designate names of G-Range components). State variables are shown in Appendix A, with their meanings briefly cited. Other simulation variables are included in that

list as well. If a user requests, these variables are written to a state variable file when a simulation is complete, so that subsequent simulations may proceed without a long *spin-up* period sometimes needed in ecosystem models. The simulation variables are stored in the file in addition to state variables so that they may be read-in in subsequent simulations, and checked against the simulation parameters of the new simulation, to ensure files are in agreement. For example, in a simulation that uses an existing state variable file, the landscape cell resolution variable in the state variable file is compared to the resolution in the current simulation. If the resolutions disagree, the simulation will produce an error and stop.

There are several scales at which G-Range functions or is organized. At the broadest scale, G-Range is a global model. In the *beta* version, all simulations include a gridded landscape bounded by -180 to 180 degrees east-west and -90 to 90 degrees north-south. The landscapes are composed of square cells (recognizing that a 1 degree square cell, for example, is usually not equal in the east-west and north-south directions when the cell is projected onto the earth's surface). A surface used by G-Range divides the globe into two land types, land and ocean. Another surface divides the cells classified as land into many land cover types, 97 types in the baseline application. A parameter file includes a list of these land cover types and an indication of whether or not each type is classified as *rangeland*. The cells indicated as rangeland are the only cells for which G-Range will simulate ecosystem processes.

Groups of pixels that are considered homogeneous in their vegetation parameters are called *landscape units* in G-Range. The globe is divided into 15 ecosystem types in the baseline application. Each of these ecosystem types has a corresponding set of parameters in the main parameter file. Many more landscape units may be included in a simulation, if the user wishes.

G-Range simulates ecosystem processes at the scale of individual landscape cells, those that are classified as rangelands. The area of land represented by each cell, in geographic coordinates, identifies the resolution of the surface. In the software and supporting files accompanying this manual, there are six sets of input spatial layers, at progressively finer resolutions. Those are 1.0, 0.5, 0.25, 0.167, 0.1, and 0.083 degrees. Geographic degrees vary in distance when projected to the earth, but for reference, at the equator, 1.0 degree latitude is approximately 111 km. At 0.083 (one-twelfth) degree resolution, a cell represents about 9.3 km. In the beta G-Range package, supporting spatial layers such as slope, aspect, and land cover type are provided at the resolutions shown. Climate data from 1901 to 2006 (CRU 2008) are provided at resolutions 1.0, 0.5, and 0.25; finer resolution surfaces are quite bulky. The remaining resolutions may be provided upon request.

Within landscape cells, G-Range divides ecosystem attributes symbolically at finer scales. Soils are simulated using four layers, in the baseline simulation set to be 0-15 cm, 15-30 cm, 30-45 cm, and 45-60 cm. Three kinds of vegetation are simulated, with those groups called *facets*,

using the terminology and some basic architectural traits of SAVANNA. Those facets are herbaceous (or herbs), shrubs, and trees. Within those types, vegetation layers are represented, including 1) herbs within the herbaceous facet, 2) herbs under shrubs and 3) shrubs in the shrub facet, and 4) herbs under trees, 5) shrubs under trees, and 6) trees in the tree facet. The extent of a given facet is defined by its overstory vegetation type, so that the herb facet includes only herbs, the shrub facet includes all shrubs, some of which may have herbs in their understory, and the tree facet is defined by trees but may include shrubs and herbs in the understory. Herbaceous plants are divided into a perennial and annual portion, and shrubs and trees include seasonal or drought deciduous and evergreen portions. Lastly, herbaceous plants are divided into leaves and shoots plus fine roots. Woody plants are divided into five parts, following Century. Those are leaves (and shoots), fine branches, coarse branches, fine roots, and coarse roots.

Process overview and Scheduling

The scheduling of G-Range is captured in the main program module shown in Appendix B. G-Range begins with initialization. First, general parameters used in the model are read, including the attributes of the simulation and locations of files needed (in *Initialize_Parms*, see Appendix B). Next, a suite of parameters are read in that describe the ecosystem processes for each landscape unit represented in the global simulation (*Initialize_Landscape_Parms*). A series of global spatial layers are then read in, describing things such as soil attributes, vegetation types, landscape unit identifiers, and tree cover (*Initialize_Globe*). From a list of vegetation types considered rangelands and the global maps, the next module creates a list of landscape cells that are rangelands, and initializes their conditions, such as field capacity, the proportion of each cell occupied by herbs, shrubs, and trees (i.e., facet proportions), and initial carbon concentrations (*Initialize_Rangelands*). Lastly, the many output files produced by G-Range are initialized, opening the files and placing in them appropriate headers (*Initialize_Outputs*).

The loops representing the monthly time-step used in G-Range are simulated, in a series of two loops, one for the years modeled, and inside that, the 12 months modeled. Following the yearly loop, a module runs that executes processes needed annually, which include setting annual accumulators back to zero, and adjusting the allocation of carbon and lignin in plant parts based on current biomasses (*Each_Year*). The monthly loop then begins, with a report of the progress of the simulation written to screen (*Progress*). Weather spatial layers are then read for the month, including precipitation, and maximum and minimum temperatures for the globe (*Read_Weather*). Spatial layers that describe management or fire are then read in, if the user has indicated that they should be used (*Read_Other*).

The main ecological processes are then simulated, within a loop that steps through the rangeland cells included in the simulation. (Calling each procedure thousands of times may appear inefficient, but this looping structure was adopted to ease the task of future parallelization of the G-Range code. Individual processors would be asked to simulate changes on a subset of

rangeland cells.) First, some attributes and indices of vegetation are updated, such as leaf area indices, carbon-nitrogen ratios, phenology based on heat accumulation, and tree basal area (*Update_Vegetation*). Weather for rangeland cells is updated next, including storm flow, heat accumulation, potential evapotranspiration, day length, and snow accumulation (*Update_Weather*).

The potential production of plant facets is calculated for each rangeland cell (*Potential_Production*). This potential is based on aspects such as soil temperature, precipitation, soil water availability, and shading. Potential production is then restricted based on nutrient limitations and nutrient ratios, for herbs (*Herb_Growth*), shrubs, and trees (*Woody_Growth*) to yield actual growth. Growth is distributed among plant parts based on allometrics. Removal of plant matter from grazing is then simulated (*Grazing*). The death of plant parts is then simulated (*Plant_Part_Death*), followed by the death of whole plants (*Whole_Plant_Death*). Land management, such as fertilization of landscapes, is then simulated (*Management*). Plant reproduction is simulated, with plants being established in accordance with several constraints (e.g., seed production, soil moisture, woody cover, herbaceous cover) (*Plant_Reproduction*). Metrics and indices of vegetation are recalculated (*Update_Vegetation*, as cited above). Water loss through transpiration, evaporation, and base flows is calculated (*Water_Loss*), then decomposition within the rangeland cell is simulated (*Decomposition*). The last set of processes simulated for the rangeland cells is for nitrogen leaching and volatilization (*Nitrogen_Losses*).

After processes for the rangeland cells are complete, the vegetation indices are updated a final time for the month (*Update_Vegetation*), then processes that must be done each month are simulated (*Each_Month*). Primarily this entails testing to see if state variables are mistakenly below zero, or extremely large values, and if so, resetting the variables to more appropriate values and tallying the number of times resetting is required (see below). Lastly, output surfaces are produced (*Output_Surfaces*). That concludes the monthly and yearly looping in G-Range. The final module called (*Wrap_Up*) produces the files that report the number of times throughout the simulation that each rangeland cell had to have its state variables reset to values above zero or below a very large number, and a report on the runtime of the simulation. The program then provides a completion statement.

G-Range Details

Inputs and Outputs

Parameter files used by G-Range were kept to a minimum, although the parameter file that describes ecosystem functions within landscape units for the globe may conceivably contain thousands of lines. The descriptions of the parameters in the files themselves are more definitive than what is shown here, and should be referenced if there are questions.

A single-line parameter file occurs in the binary executable folder of G-Range (i.e., *G_Range_Bin*) called **GRRange.ini**. The parameter shows the pathway to the current G-Range application folder:

```
'C:\G_Range' // The pathway to the application (no trailing slash or backslash)
```

This file may be edited if the G-Range software is installed on a hard drive partition with a different designation. It may also be edited to maintain parallel G-Range applications (although there are more typical means to conduct scenario analyses describe below). For example, a user may maintain a copy of G-Range under “C:\G_Range_Normal” and other under “C:\G_Range_Extreme”. (Note that the development folder for G-Range is GRRange, so the pathway provided with the beta version of G-Range is C:\GRRange\G_Range rather than the shorter pathway shown above.)

Within the folder shown in **GRRange.ini**, G-Range is hardwired to look for parameters in the *Parms* folder. The main parameter file is hardwired as well, and called **Sim_Parm.grg**. That parameter file includes a flag that indicates if many of the inputs into G-Range are to be echoed to an output file called **Echo.gof**. For the fullest echoing of input data, that flag should be set to TRUE. Another parameter controls some echoing to the screen. A lengthy series of parameters provide pathways and file names to the spatial layers used by G-Range to describe attributes of landscape cells. The spatial layers used by G-Range are described below. The pathways and names of the spatial layers may be changed by editing this parameter file, to represent different scenarios. For example, a user may store future weather data from different atmospheric-ocean general circulation models under different folders (e.g., ‘\layers_hadley\’ and ‘\layers_ncar\’), and switch between the weather data sets by editing **Sim_Parm.grg**. An entry in **Sim_Parm.grg** includes the name of a file (**goge.leg** in the base simulation) that contains the legend to a spatial layer showing land cover types of the globe. Its purpose is described below. Given that hundreds of weather spatial layers may be used in a simulation, rather than providing specific spatial layer names, a different approach was used to indicate their names. The prefixes of precipitation, maximum temperature, and minimum temperature are indicated in **Sim_Parm.grg**, together with their suffix, which is the same across the three weather map types. For example, in the base simulation, the prefix to the precipitation layers is shown as ‘\layers\cru_prcp\p’ and the weather layers that are at 1.0 degree resolution share the suffix ‘_1p0.asc’. During a simulation, G-Range creates a file name for each weather file based upon the prefix, suffix, and the month and year being simulated (e.g., ‘\layers\cru_prcp\p197501_1p0.asc’). **Sim_Parm.grg** includes an entry providing the name for the main landscape unit parameter file (**Land_Units.grg** in the base application). A series of parameters provide the pathways to spatial surfaces of fire and fertilization histories, parameter files describing those histories cited below, and switches that show whether fire or fertilization should be managed using the spatial surfaces or values in the landscape unit parameter file. That is, using fire as an example, proportions of landscape cells burned may be explicitly scheduled using spatial surfaces, or fires may be included probabilistically by setting values in the

parameter file *Land_Units.grg* and turning off the use of fire spatial surfaces. Parameters include the pathway and file names of state variable files (e.g., '*output\state_base_out.grg*' and '*output\state_base_in.grg*'), which are used to store the state of a simulation at its completion. That file may then be used to begin another simulation. This is done to avoid long 'spin-up' periods when simulation models are reaching equilibrium. A spin-up simulation is done and results are written to a state variable file, then that file may be used in successive simulations. A switch controls the behavior of G-Range in that regard. Following SAVANNA, a value of 0 means that a state variable file is not used, 1 means that the file is written at the end of a simulation, 2 means that a state variable file is used to initialize G-Range, and 3 means that the state variable file is both used to initialize, and written at the end of a simulation. Lastly, **Sim_Parm.grg** includes the starting and ending years of the simulation.

A spatial layer used as input into G-Range to define rangelands and described below includes the accompanying parameter file called **GoGe.leg** in the base application, but the name may be changed in **Sim_Parm.grg**. The structure of the file is simple, with a value showing the number of land cover types included in the land cover spatial layer, then that number of entries, with one for each land cover type. Each entry includes a switch showing whether the land cover type is to be considered rangeland (1) or not (0), a land cover numerical identifier, and a land cover name. A user may change the subset of global cells that are considered rangelands by changing the switch of a land cover type. Moreover, a user may replace the entire spatial surface if they wish to use another to define rangelands.

Distinct from identifying rangeland cells, another spatial layer is used in G-Range to identify the landscape units that are to be considered homogeneous in their basic ecosystem functions. A parameter file (**Land_Units.grg** in the base application) provides the suite of parameters that allows G-Range to simulate global rangelands, with plants and soils in different landscape units having unique attributes. In the base application, the globe is divided into 15 potential vegetation types, following Conant et al. (in review), a subset of which include rangelands. The general structure of **Land_Units.grg** is that it is composed of groups of 75 parameters with each representing a given landscape unit. The first 75 parameters are followed by another set for the next landscape unit, continuing until all units are described. A user may divide land masses of the globe into dozens, hundreds, or more landscape units in an application, limited only by the degree to which working with the landscape unit parameter file becomes unwieldy. The 75 parameters typifying a landscape unit in **Land_Units.grg** are reviewed in Appendix C. Users may maintain multiple versions of landscape unit parameter files. The main means to conduct scenario analyses will be either in editing a file analogous to **Land_Units.grg**, or keeping multiple versions of that file (e.g., **Land_Units_Base.grg**, **Land_Units_High_Fire.grg**, **Land_Units_Invasives.grg**) and changing the file used by editing **Sim_Parm.grg**.

Two parameter files are similar, providing the names of spatial layers that control fire or fertilization in rangelands across the globe. The use of those files is controlled by changing a switch in **Sim_Parm.grg**. If the switch is set to 0, fire and fertilization is controlled through parameters in the main landscape unit parameter file, the **Land_Units.grg** file in the base application. (In that parameter set, frequencies of fire or fertilization may be set to 0, turning off those modeling components so that fire or fertilization are not simulated.) If the switch is set to 1, then the pattern of fire or fertilization is controlled by spatial layers, listed in the parameter files that are the focus here. Those files are called **Fire_Dat.grg** and **Fert_Dat.grg** in the base application, although those names may be changed in **Sim_Parm.grg**. Those parameter files are straightforward, with as many lines as needed showing the year the spatial layer is to begin being used, the month the spatial layer is to begin being used, and the name of the spatial layer. Note that the pathway to the spatial layers is provided in **Sim_Parm.grg**. The file ends with end-of-file markers of -99 and '-99'. An example **Fire_Dat.grg** would be:

```
1960, 1, 'no_fire_1p0.asc'           // Year, month, and map name
1963, 8, 'many_fires_1p0.asc'       // Year, month, and map name
1963, 9, 'no_fire_1p0.asc'         // Year, month, and map name
1964, 6, 'spotty_fire_1p0.asc'     // Year, month, and map name
-99, -99, '-99'                    // End of file markers
```

A user may sense that **Fire_Dat.grg** may be very lengthy for long simulations, but very easy to understand and use. This reflects a point of view of G-Range, that clarity is valued over compactness. In the example, a spatial layer without fire is used beginning January of 1960, and then in August of 1963, many fires are simulated. The next month returns to no fires being simulated, then in June of 1964, fires in a spotty pattern are simulated. That spatial layer will be used until the end of the simulation, whenever that may be. The nature of the spatial layers is described below. As in other cases, multiple copies of fire and fertilization parameter files may be stored, and the name of the files used changed in **Sim_Parm.grg** to conduct scenario analyses.

The final parameter file is called **Output_List.grg**, a name hardwired within G-Range.

Output_List.grg contains information useful to a GUI, describing the structure of output files. One entry is suitable for editing by users, however. The file includes more than 100 entries, one for each output file. Two typical lines of **Output_List.grg** would be:

```
1, 'bare_cover', 1, ' ', 'Proportion', 'Bare ground for the cell'
1, 'facet_cover', 3, 'Facet', 'Proportion', 'Proportion of cell per facet'
```

The first column may be edited by a user. If it contains a 1, that output file is created. If the column contains a 0, the file is not produced during a simulation. If a user finds an output type to not be useful and wishes to conserve space on the storage media, that output may be disabled. The remaining columns provide the name of the output file, the number of spatial surfaces per month stored in the file, the name of the surfaces if different than 1, units, and a description. In

the example given, the 1s in the first column means that both kinds of maps will be produced. Files called **bare_cover.gof** and **facet_cover.gof** will be formed, as shown in column two. The third column indicates that there is a single layer in the bare cover output file; every landscape cell has just a single value for bare ground. In contrast, facet cover includes a spatial layer for herbs, another for shrubs, and a third for trees every month, all stored within **facet_cover.gof**. The fourth column labels these layers as facets. Both files store data that represent proportions, as shown in the fifth column, and lastly, a brief description is given.

The other inputs into G-Range are a series of spatial layers. These are in GRIDASCII format, a straightforward format originating with ESRI® products such as ArcGIS and produced and used by a wide variety of geographic software. The spatial layers have an intrinsic resolution, with the square cell size shown as an entry near the top of each GRIDASCII file. Spatial layers in G-Range are in geographic coordinates (i.e., they are not projected), but the GUI supplied with the software will project the results if desired. The resolutions supplied with the base G-Range application are 1.0, 0.5, 0.25, 0.167, 0.1, and 0.083 degrees, with the finest resolution weather data available on request. Spatial layers describe attributes of the cells that include spatial location, a cell identifier, soil attributes (silt, sand, clay, gravel, bulk density, organic carbon), vegetation attributes (herbaceous, shrub, and tree cover, deciduous and evergreen tree proportions), plus weather data. The spatial layers used by G-Range are described in Appendix D. The sources of those data used in the baseline G-Range application are cited in footnotes in that appendix. Note that the precipitation and temperature spatial surfaces are multiplied by 10 in the input files, to retain precision but allow storage of integer GRIDASCII files, which are compact.

A small set of additional output files are produced by G-Range. A lengthy text file is produced called **Echo.gof**. That file contains free-form output that echoes input into G-Range. The echoing of spatial layers may be turned on or off in **Sim_Parm.grg**. Input values for G-Range simulation parameters are echoed in full, followed by the parameters controlling ecosystem process in each of the landscape units. Spatial layers are echoed if requested, with the values for cells in the spatial layers converted using divisors defined informally so that each cell was represented by a single digit, yielding well-formed spatial representations. As spatial layers for precipitation and temperature are read in during a simulation, their reading is echoed to **Echo.gof**. Programmers compiling G-Range may request that various debugging information be written to **Echo.gof** as well.

Globe.gof is a simple text file that contains a matrix of letters with the numbers of rows and columns equal to the dimensions of the spatial layers. Each row represents cells, and contains a T or an F, depending upon whether the cell represents land (T) or water (F). Having the graphical user interface produce a map of disjointed groups of rangeland cells would not be

helpful. The **Globe.gof** file is used by the GUI to provide a global context to the rangeland cells simulated, allowing the GUI to paint land a different color than water.

A file is produced that informs the user, but is primarily used by a graphical user interface, about the cells in the global spatial surface that are classified as rangeland. The file, called **Rng_Data.gof**, has a header line that reports the number of rangeland cells defined for the simulation, the width of the spatial layers used in the simulation in number of cells (e.g., 360 for a 1 degree spatial surface), the height of the spatial layers in number of cells (e.g., 180 for a 1 degree surface), the lower left coordinate for the surfaces (-180 and -90, given that the surfaces are global), the upper right coordinate for the surfaces (180 and 90), the dimension of the cells (1.00000 in a 1 degree surface), the beginning year of the simulation, and the ending year of the simulation. For each rangeland cell, a line follows the header that provides the index of the cell (i.e., 1 to rangeland cell count), a zonal identifier, which is a unique value given to the cell, the position of the cell in X and Y spatial surface units (i.e., reporting that a cell occupies column 121 and row 83, rather than in geographic coordinates), and the rangeland type of the cell.

G-Range is simulating thousands of rangeland cells. An error in a single pixel should not halt an entire simulation; such behavior would make debugging an application extremely difficult and time consuming. Instead, we incorporated a module that tests rangeland cells as to whether they are below zero and shouldn't be, or are extremely large and shouldn't be (e.g., such as would happen if a divide by a miniscule value had occurred). One may wish to report a spatial surface for every output file type that included the number of times it had exceeded these limits, but that would swell the already voluminous output from G-Range. Instead, we produce a text file reporting the number of errors non-spatially for each output type, and spatial surfaces that combine all output types, allowing a user to see if errors are clumped spatially. **Exceed.gof** is a textual output file that reports the number of times cells for each output type that should not go below zero did indeed go below zero, and the number of times cells for each output type exceeded a very large number (i.e., 1,000,000,000), threatening to exceed what computer programmers call *maximum integer* and cause an error. The format of the file is simple, with a descriptive header line, followed by an entry for each output file, a count of the number of times the value went below zero, and a count of the number of times the value became very large. If a given value for a cell was below zero, it was reset to zero and the error tallied for ultimate output to **Exceed.gof** and the spatial surface **neg_error_count.gof**. Similarly, if a value for a given cell exceeds a very large number, it is reset to that large number and the error tallied in **Exceed.gof** and **large_error_count.gof**. A user may consider these three files together to get a good impression of the types, degrees, and locations of errors occurring during a simulation. When interpreting **Exceed.gof**, recall that it is tallying every time an output value exceeds a limit. In the base simulation at 1.0 degree, for example, there are 7439 rangeland cells. A 10 year simulation would include almost 900,000 opportunities for an error (i.e., 7439 x 12 x 10) to occur in any output type. Values within **Exceed.gof** should be interpreted in light of this.

Run_time.gof is the last, minor, output file. It is a text file that reports the time required to complete the simulation, plus the time required to complete a standardized portion of the simulation. The report includes the seconds, minutes, and hours required for completing the simulation, the number of rangeland cells simulated, years modeled, the seconds required to simulate a cell-month, and to simulation 1000 cell-months.

G-Range Procedures and Subroutines

Detail on select portions of G-Range are provided in Append F. For more explanation on plant-soil nutrient cycling, readers may refer to the manual provided with Century 4.0 (Metherell et al. 1993). Documenting G-Range to the degree included in the Century 4.0 manual is beyond the scope of this introductory report. Note that Century 4.5 was used as reference when coding G-Range, and so the material in the manual for Century 4.0 may be in some disagreement. There have been simplifications made when streamlining Century for use in G-Range. For example:

- Phosphorous and sulfur are not traced in G-Range;
- Forests and agricultural lands are not modeled separately in G-Range. In general, what are referred to as crops in older portions of Century code, and material that referenced savannas in newer Century code, were referenced while coding G-Range;

That said, the manual for Century 4.0 will provide extensive information that does apply to G-Range.

The G-Range Graphical User Interface

Our focus has been on creating, assessing, and using the G-Range global rangeland simulation tool, rather than in creating a highly developed and portable GUI. Creating such an interface remains of interest (see Future Steps). But we needed a GUI that was straightforward to use with G-Range, to allow output to be viewed and exported. Exported spatial surfaces may then be analyzed by tools of a user's choosing, outside the G-Range GUI. The GUI for use with G-Range ("G-Range GUI," or GRange_GUI.exe) is summarized here.

GRange_GUI.exe is a Windows compatible executable created in Windows Visual Basic 6®. The tool needs to be installed on a computer prior to use, which may be done by running **Setup** in the distribution media under G:\GRange\G_Range_GUI\Package. If a user does not wish to run Setup, the OCX control called MSCHRT20.OCX accompanying the software may be registered (i.e., Windows Start menu -> Run -> REGSVR32 MSCHRT20.OCX, with a path to the OCX file if required). GRange_GUI.exe should then be executable.

Two forms comprise G-Range GUI, a large main form used to explore G-Range output, and a form that is opened when a user is exploring data. Examples of the forms are shown in Figures 2 and 3, with numbers overlaid to identify elements.

GRange_GUI uses the same one-line **GRange.ini** parameter file stored within the GRange “bin” folder to learn where the main application folder is located (e.g., “C:\G_Range” or “C:\GRange\G_Range” in the beta package). Within that folder, GRange_GUI expects a “Parms” folder which stores information about the output files produced by GRange (**Output_List.grg** described above) and an “Output” folder with its complement of output files.

The GUI begins with a blank main window (“1” in Figure 1). When a user clicks on an output file type within the scrollable window of files (2), a map of the output from the first month of the simulation may be produced. A user may then select from the different symbolizations provided (3). For continuous data, stretched red, green, or blue responses are appropriate, or the stretched blue to yellow symbolization is effective and is the default. Regardless of the color of the stretched symbols, their minimum value is based on the mean minus twice the standard deviation of the values within the spatial data being mapped (rather than across the entire set of maps of that type) – any pixel data in the spatial layer that is equal to or lower than that value receives the darkest color included in that symbol set. The maximum value for the stretched symbols is defined as the mean plus twice the standard deviation, and any pixel in the layer with a value greater than or equal to that result will receive the brightest symbol. Other symbols are stretched linearly between the extremes. The other symbolization option is for random colors, which is appropriate for categorical information such as range type. The colors are random, but stable between uses of GRange_GUI.

After accepting or selecting a symbolization set, a user may then scroll through different months and years of the simulation (4), with the map updating with each new time period. If months are being changed and 12 is exceeded, the year will change to the following year and the month to 1. If the end of the simulation is reached, a warning tone is provided and the last year is displayed again (i.e., the year does not change, but month returns to 1). Analogous behavior occurs if stepping from newer to older months of a simulation. Lastly, the “Auto” button (4) may be selected to automatically step through months of a simulation at a constant rate. The user may select another output type (2) directly, and the month and year shown (4) will be displayed.

A user may click on an output type (2) and find that the map does not display. This is because the file contains spatial data that includes multiple layers. For example, a user who clicks on “bare ground” from the list of output files will be shown the results from the first month of the simulation; there is single value for percent bare ground for any landscape pixel, and so no ambiguity. If a user selects “facet cover,” that file contains output for the herbaceous (facet 1), shrub (facet 2), and tree facets (facet 3). A window in the form (5) is populated with the layers within the output file type. When the user clicks on one of those entries, a map is produced.

Summary statistics are provided for the spatial layer being mapped (6), including the units used and a reminder of the number of rangeland cells modeled. When a user clicks on a location in

the map, the values through the simulation for the nearest rangeland cell are plotted (7). Note that the nearest cell is used, so if a cell in the ocean is selected, a cell that may be somewhat distant from the cursor will be plotted. The plotted cell is painted yellow. To put responses in some context, precipitation, minimum temperature, and maximum temperature for the cell are automatically plotted on a secondary axis (7).

The GUI for G_Range will display cells in magenta if a stretch cannot be performed on them (1). This typically means that the mean and standard deviation are both zero, and the surface contains all zeros (6). This is normal behavior. For example, the first month of simulated heat accumulation is at zero for all rangelands, and so the rangelands of the globe are shown in magenta. Moving to the next month produces a map displayed reasonably. That said, G-Range may produce values in simulations that are beyond the range expected by GRange_GUI. Those exceptions have not yet been handled, and the software stops functioning and must be restarted.

If a user wishes to export surfaces from G-Range, they would use that button on the main form (8). That will open the export form (Figure 2). A user would select an output folder they would like the files created to be written to (9). The Windows folder selection tool may be used (using the “Select folder” button) or the folder may be typed in. The user then provides the root name for the surfaces, with the name of the output file provided as a suggestion. That name may be edited as the user wishes to reflect different scenarios, or the output folder may be changed between scenarios. The years to be output are selected (11); the span of the simulation is selected by default, and a button is provided as a shortcut to select all years. The number of decimal places to be used in the output GRIDASCII files may be set. The tool defaults to three decimal places. Lastly, the projection to be used in output may be selected, with four choices available, unprojected geographic coordinates plus three global projections. Briefly, attributes for the geographic and projected spatial layers are:

- Projection: Geographic, Units: Decimal degrees, Datum: WGS84, Spheroid: WGS84, XShift: 0.000, YShift: 0.000
- Projection: Robinson, Units: Meters, Spheroid: Sphere, XShift: 0.000, YShift: 0.000, Longitude of central meridian: 0.0, False easting: 0.000, False northing: 0.000
- Projection: Eckert IV, Units: Meters, Spheroid: Sphere, XShift: 0.000, YShift: 0.000, Longitude of projection center: 0.0
- Projection: Winkel Tripel, Units: Meters, Spheroid: Sphere, XShift: 0.000, YShift: 0.000, Radius of the sphere of reference: 6370997.000, Longitude of central meridian: 0.000, Latitude of standard parallel: 50 28 1.200, False easting: 0.000, False northing: 0.000

Using a projection during exporting can provide spatial surfaces that are less distorted, where areas, shapes, or distances are better preserved than when in unprojected geographic coordinates. After selecting the projection to be used, the user may select “Go” (14) and the surfaces are exported in GRIDASCII format. “Cancel” dismisses the Export form.

An Example Scenario using a Changing Climate

A baseline application was created that captures the broad-scale responses in rangeland water relationships and plant production. Figure 4 compares average aboveground live biomass weighted by facet area in 1988 with total net primary productivity from Del Grosso et al. (2008). The metrics used are not directly comparable, but are related, and generally, the broad patterns are found. Figure 5 shows a correction factor applied to decomposition rates in both models. The measures are not directly comparable, but the overall patterns are similar. Further adjustments to parameters in **Land_Units.grg** would presumably improve the agreements. Note that broad patterns only were compared to expectations. This application still needs to be assessed (see Future Steps), and should not be used in research or demonstrations. Our focus has been on the structure and functioning of G-Range, leaving insufficient time to parameterize a global application with great confidence.

Scenario

To demonstrate a scenario analysis with G-Range, we used a set of monthly weather surfaces provided by Dr. Philip Thornton and developed by the Climate Change, Agriculture and Food Security Program (<http://www.ccafs-climate.org>) to which he is affiliated. We selected from the data set three layers of information from two scenarios. Those include precipitation, minimum temperature, and maximum temperature from 2037 to 2066, for scenario A2 from the IPCC report (IPCC 2007) and scenario B1. From the IPCC (2007) summary, “B1 describes a convergent world, with the same global population as A1 [a global population that peaks in mid-century], but with more rapid changes in economic structures toward a service and information economy,” and “A2 describes a very heterogeneous world with high population growth, slow economic development and slow technological change.”

The data were reformatted into GRIDASCII spatial layers for use in G-Range, then **Sim_Parm.grg** was edited to point to the new surfaces. Simulations were made at 1 degree resolution for 2040 to 2060 for both scenarios.

Results

Selected results from the simulations are shown in Figures 6 and 7. Primarily these results demonstrate that the G-Range infrastructure is in place and working reasonably – the specific results should not be cited, pending a thorough calibration of G-Range to a baseline scenario and anticipated effects of climate change. Figure 6 compares shrub cover (facet 2) for a particular cell in Asia, for the same date. Figure 7 compares tree leaf area index for an area of central Canada. For that area of the globe, leaf area index is greater under scenario A1 after a 20 year simulation that includes future climate change projections.

Future Steps

G-Range is a functioning and reasonable global simulation tool of moderate complexity. A main means of debugging and enhancing the simulation tool will simply be through its use to address new questions. That said, as a beta application, many enhancements are possible. These include:

- G-Range requires further testing and debugging. That said, many of the components of G-Range have been assessed over many years of Century simulation. Any error reports should be forwarded to the senior author of this report at the address listed on the title page, and related to that;
- The baseline simulation provided with the beta version of G-Range has not been assessed effectively. A more thorough exploration of the output from G-Range than could be done given time limitations is required. The same is true for the output from simulations using future climate. They have not been explored well. Also, information used to compare against is not readily available, for rangelands in general. Global modeling of rangelands only is not common, and comparisons are not trivial. That said, output from specific cells from Century runs may be compared to output from the same cells in G-Range output, to build confidence in the new model. Rangeland dynamics have not yet been simulated at high spatial resolutions. Presumably minor complications will follow from that;
- The baseline simulation and spatial data include a few cells that do not align correctly. For example, this is where something classified as rangelands in one layer may be classified as ocean in another. They are generally on the edges of land masses. The cells that disagree cause an error to be reported in **Echo.gof**. The layers should be corrected by a GIS technician.
- G-Range should be exercised more than has occurred prior to publication of this report. Limitations on time and other constraints have prevented us from conducting simulations at fine spatial scale, sensitivity analyses, backcasting of simulations, etc.
- An addition that may be made to G-Range is a set of products that integrate the fine-scale representation of individual plant properties and broad-scale (1 km²) representation of plant populations. For example, a desert with moderately productive plant that had very low cover would show average net primary productivity across the landscape as quite low. This kind of analysis yielded Figure
- The simulation platform may be quite helpful at resolutions well below global simulations. Making G-Range work for only regions of the globe would be straightforward, and is an early next step for its development. Until that occurs, users may wish to turn off rangelands not to be modeled in areas of the globe not of interest to them;
- Herbivore dynamics should be represented in G-Range. We have the expertise to do this, and have included such dynamics in another model (Boone et al. 2011);

- Decomposition may be tracked across soil layers, rather than for only surface and soil layers as in the current version. In the current version, nitrogen and water flow through the soil layers, but carbon does not;
- A graphical user interface should be programmed in Java to allow it to be as portable as the G-Range simulation program. Such a GUI could be used on Windows, Linux, Unix, Macintosh, or other platforms and would yield the same results;
- Many improvements to the graphical user interface may be envisioned. Important among those is the ability to compare the results of two or more simulations;
- A graphic user interface may be created, again in Java, which eases and manages the creation of the landscape unit parameter file (i.e., **Land_Units.grg** in the base application).
- A GUI that simplifies parameterization of what is now called Land_Units.grg, including definitions, etc. In the base application, the globe was divided into 15 zones of potential vegetation, but users may wish to divide the globe into many more. A software tool that supported that effort would be useful. Such a tool would display parameters for each landscape unit separately, provide definitions for the parameters, and write landscape unit parameter files known to be formatted correctly;
- A web site may be developed that describes how to use G-Range and makes the simulation tool available.

Acknowledgements

Our sincerest thanks to the developers of Century. We especially thank Dr. William Parton for being supportive of our efforts to use Century as a foundation for G-Range, and Cindy Keough for her patience with questions. Our thanks to Dr. Michael Coughenour as the author of SAVANNA, from which some ideas for G-Range were drawn. We thank Drs. Mario Herrero and Philip Thornton and the International Livestock Research Institute for providing financial support for the creation of G-Range.

Literature Cited

- Boone, R.B., K.A. Galvin, S.B. BurnSilver, P.K. Thornton, D.S. Ojima, and J.R. Jawson. 2011. Using coupled simulation models to link pastoral decision making and ecosystem services. *Ecology and Society* 16(2):6 [online] URL: <http://www.ecologyandsociety.org/vol16/iss2/art6/>
- Climate Research Unit (CRU). 2008. CRU Datasets. British Atmospheric Data Centre. University of East Anglia. Online: <http://badc.nerc.ac.uk/data/cru>
- Conant, R.T., T.E. Hilinski, A.B. Berdanier, R.B. Boone, and D.S. Ojima. In review. Volatility of forage production and the world's livestock production systems: implications for climate change adaptations. *Proceedings of the National Academy of Science*

- DeFries, R.S., M.C. Hansen, J.R.G. Townshend, A.C. Janetos, and T.R. Loveland. 2000. A new global 1-km dataset of percentage tree cover derived from remote sensing. *Global Change Biology* 6:247-254.
- Del Grosso, S., W. Parton, T. Stohlgren, D. Zheng, D. Backelet, S. Prince, K. Hibbard, and R. Olson. 2008. Global potential net primary production predicted from vegetation class, precipitation, and temperature. *Ecology* 89:2117-2126.
- Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., Goss-Custard, J., Grand, T., Heinz, S.K., Huse, G., Huth, A., Jepsen, J.U., Jørgensen, C., Mooij, W.M., Müller, B., Pe'er, G., Piou, C., Railsback, S.F., Robbins, A.M., Robbins, M.M., Rossmanith, E., Rüger, N., Strand, E., Souissi, S., Stillman, R.A., Vabø, R., Visser, U. and DeAngelis, D.L. 2006. A standard protocol for describing individual-based and agent-based models. *Ecological Modelling* 198:115-126.
- Grimm, V., Berger, U., DeAngelis, D.L., Polhill, J.G., Giske, J. and Railsback, S.F. 2010. The ODD protocol: A review and first update. *Ecological Modelling* 221:2760-2768.
- Gupta, S.C. and W.E. Larson. 1979. Estimating soil water retention characteristics from particle size distribution, organic matter content and bulk density. *Water Resource Research* 15:1633-1635.
- Hansen, M., R.S. DeFries, J.R.G. Townshend, M. Carroll, C. Dimiceli, and R.A. Sohlberg. 2006. *Vegetation Continuous Fields MOD44B, Collection 4*. University of Maryland, College Park, Maryland, USA.
- Hobbs, N.T., K.A. Galvin, C.J. Stokes, J.M. Lackett, A.J. Ash, R.B. Boone, R.S. Reid, and P.K. Thornton. 2008. Fragmentation of rangelands: implications for humans, animals, and landscapes. *Global Environmental Change* 18:776-785.
- Intergovernmental Panel on Climate Change (IPCC). 2007. *IPCC Fourth Assessment Report: Climate Change 2007 (AR4)*. Cambridge University Press, Cambridge, United Kingdom and New York, New York, USA.
- Metherell, A.K., L.A. Harding, C.V. Cole, and W.J. Parton. 1993. *Century manual: Century soil organic matter model environment technical documentation, Agroecosystem Version 4.0*. Great Plains System Research Unit Technical Report No. 4. USDA-ARS, Fort Collins, Colorado, USA. Online: <http://www.nrel.colostate.edu/projects/century/documentation2.htm>
- Monteith, J.L. 1965. Evaporation and environment. In: G.E. Fogg (ed.), *Symposium of the Society for Experimental Biology, The State and Movement of Water in Living Organisms*, Vol. 19, Academic Press, New York, New York, USA. pp. 205-234.
- NREL (Natural Resource Ecology Laboratory). 2000. *Century 4 Homepage*. Natural Resource Ecology Laboratory, Colorado State University, Fort Collins, Colorado, USA. Online: <http://www.nrel.colostate.edu/projects/century/reference2.htm>
- NREL (Natural Resource Ecology Laboratory). 2007. *SAVANNA – Readme: Publications and reports using the SAVANNA model*. Natural Resource Ecology Laboratory, Colorado

- State University, Fort Collins, Colorado, USA. Online:
<http://www.nrel.colostate.edu/projects/savanna/>
- NREL (Natural Resource Ecology Laboratory). 2010. Integrated research challenges: terrestrial carbon model. Natural Resource Ecology Laboratory, Colorado State University, Fort Collins, Colorado, USA. Online: <http://www.nrel.colostate.edu/projects/irc/>
- Peck, S.L. 2004. Simulation as experiment: a philosophical reassessment for biological modeling. *Trends in Ecology and Evolution* 19:530-534.
- Penman, H.L. 1948. Natural evaporation from open water, bare soil and grass. *Proceedings of the Royal Society of London, Series A.* 193:120-146.
- Ramankutty, N. and J.A. Foley. 1999. Estimating historical changes in global land cover: croplands from 1700 to 1992. *Global Biogeochemical Cycles* 13:997-1027.
- Rawls, W.J., D.L. Brakensiek, and K.E. Saxton. 1982. Estimation of soil water properties. *Transactions of the American Society of Agricultural Engineers* 1316-1328.

Glossary

Beta – A stage in software development. All aspects of formal definitions do not apply here; here we use the term as the first release of a software package that is available to those outside the organization that developed the software.

Facet – Following SAVANNA, in G-Range a facet is one of the main types of vegetation, either herbaceous, shrub, or tree.

Hardwired – An informal term in computer programming meaning that an entry is programmed directly into a software package, and cannot be modified by a user without re-compiling the executable file.

Landscape unit – A group of landscape cells for which biological attributes and processes are considered homogeneous. The physical nature of the cells within a landscape unit may be unique, as described in spatial layers used in G-Range. However, the nature of vegetation, plant growth, livestock grazing, etc. is homogenous across the unit. Landscape units are usually contiguous, but that is not required.

Maximum integer – A term used in computer science to refer to the largest integer that may be represented in a given compiler. This is typically an extremely large value (e.g., +2,147,483,647). In an ecosystem model such as G-Range, a value at or approaching maximum integer suggests a problem.

Rangeland – In G-Range, a simple true-false designation indicating whether or not a landscape cell will have ecosystem processes simulated.

Spin-up – A period early in a simulation used to allow state variables to reach some kind of equilibrium. Many packages, including G-Range, allow the state of a simulation to be saved after a spin-up period, then read in subsequent simulations and used as their starting points.

Figures

Figure 1. A schematic diagram showing the shrinking adaptive capacity of pastoral communities (large circle, yellow) due to constraints (outer circles) over time.

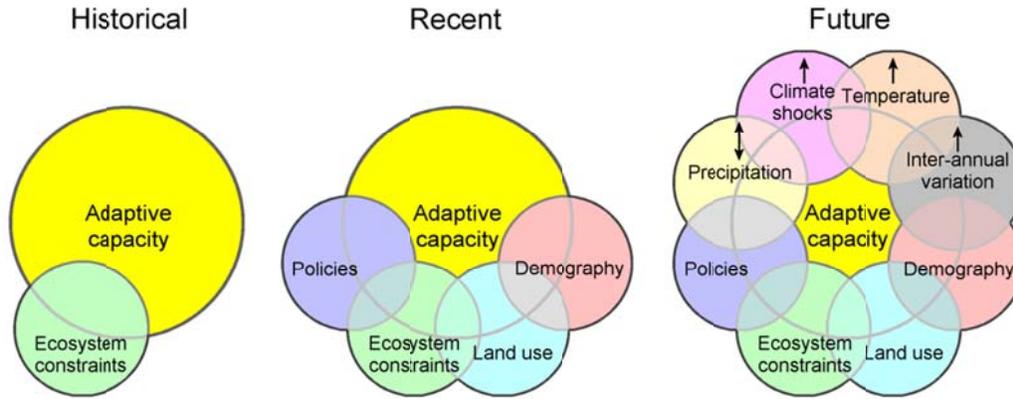


Figure 2. The main form of graphical user interface provided with the beta version of G-Range. Numbers have been overlaid on the form, and are referenced in the body of this report (see *The G-Range Graphical User Interface*).

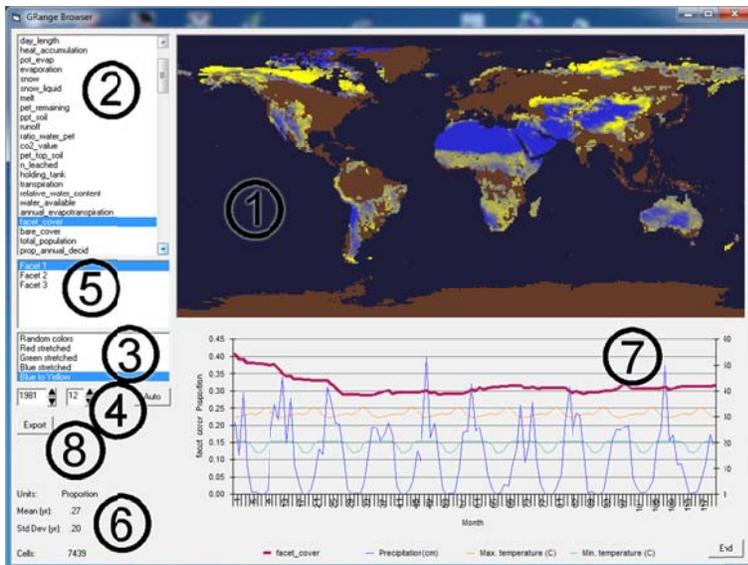


Figure 3. The export form opened when requested by a user of the graphical user interface provided with the beta version of G-Range. Numbers have been overlaid on the form, and are referenced in the body of this report (see *The G-Range Graphical User Interface*).

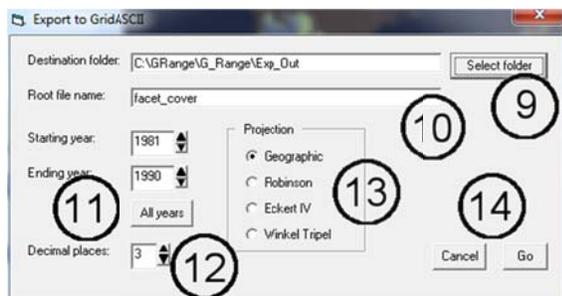


Figure 4. Average aboveground live biomass (g/m^2) per cell (a), combining the herb, shrub, and tree facet biomasses, weighted by their facet cover. The general patterns may be compared with estimated total net primary productivity (g C/m^2) from Del Grosso (2008). Although the values used to classify the images are the same, note that the units and underlying data are related but not the same.

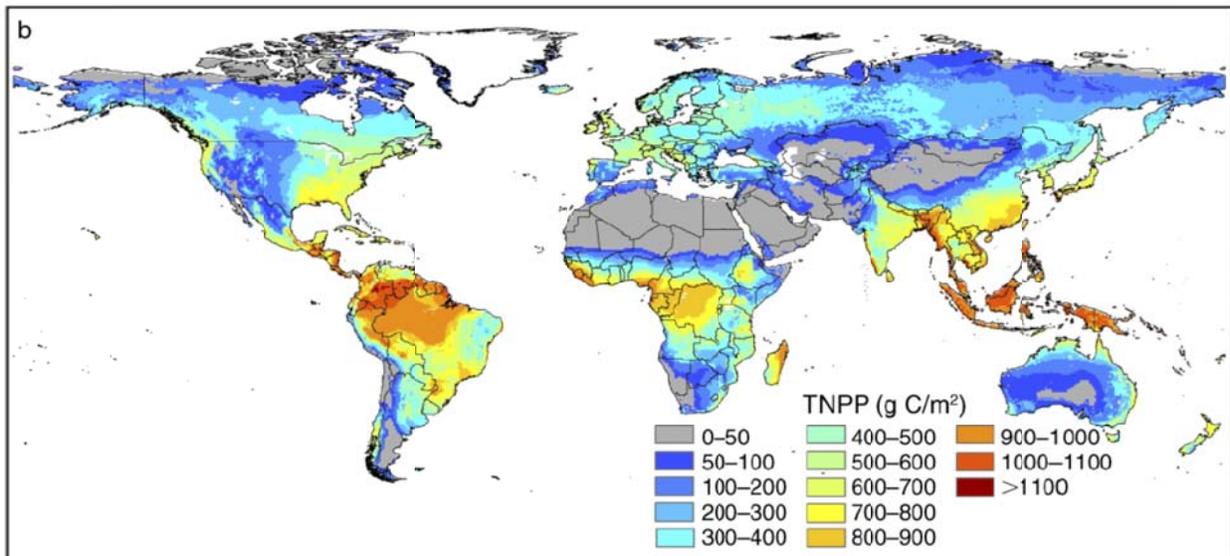
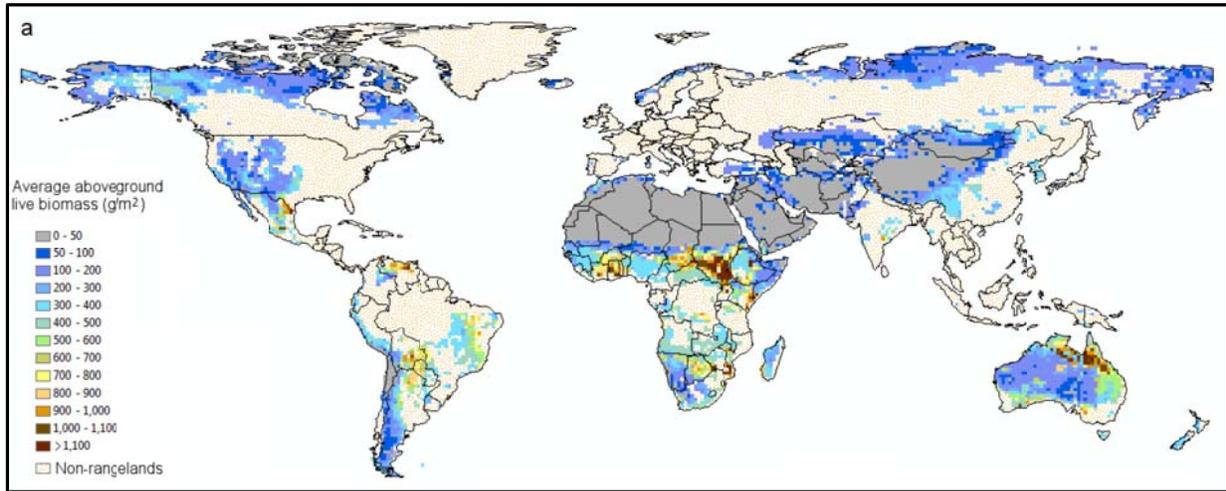


Figure 5. The combined effects of temperature, water, and anaerobic conditions on decomposition rates from G-Range, averaged across 1988 (a), compared with effects of temperature and water on decomposition in Century simulations (b) (unpublished simulation results). Attributes in the two metrics differ (e.g., anaerobic limitations included in G-Range), but the general patterns are correlated.

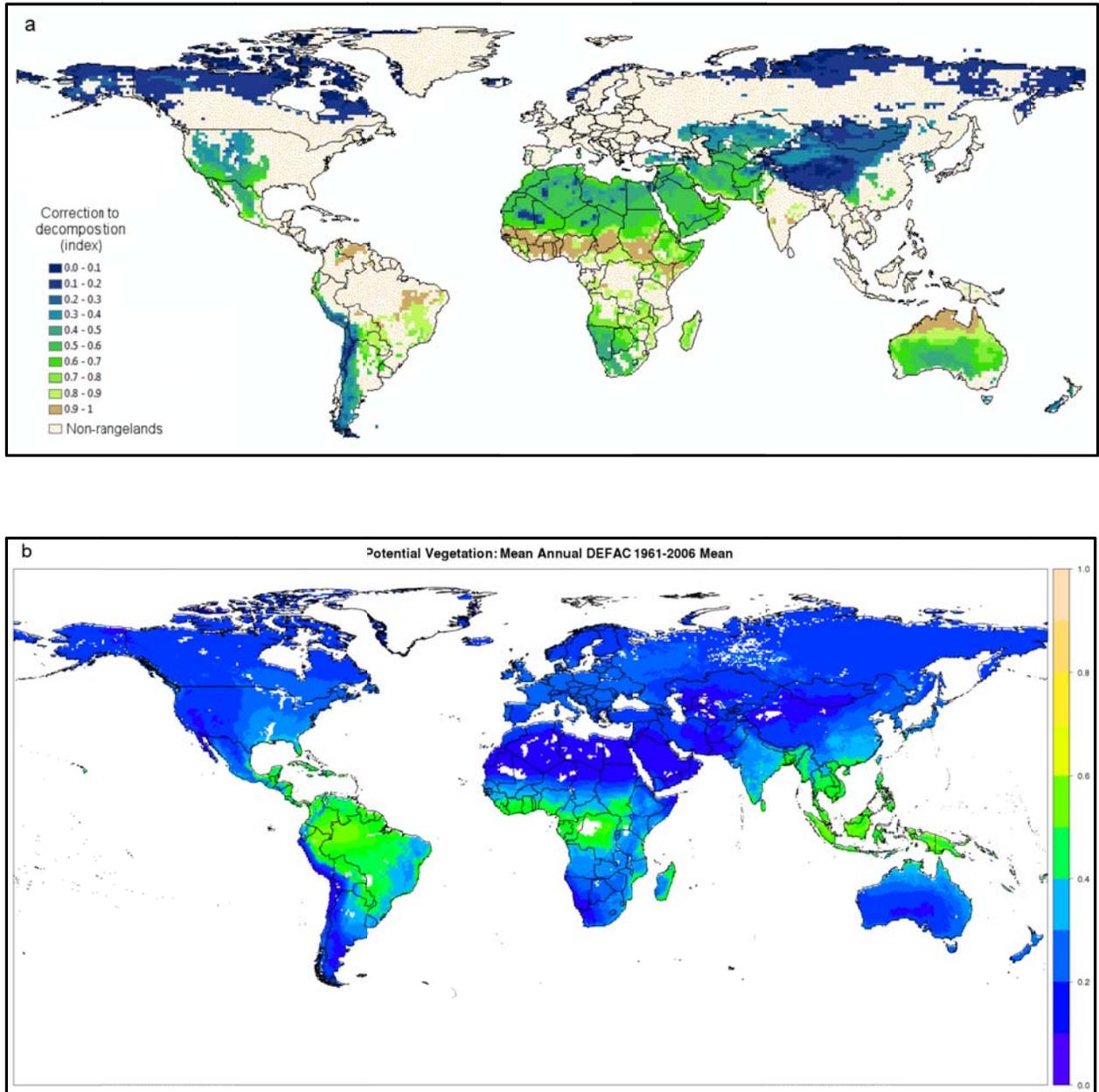


Figure 6. Comparisons of shrub cover in June 2042 under the A2 (a) and B1 (b) scenarios adopted in the IPCC (2010) reports. The graph is from a region in northwest Asia.

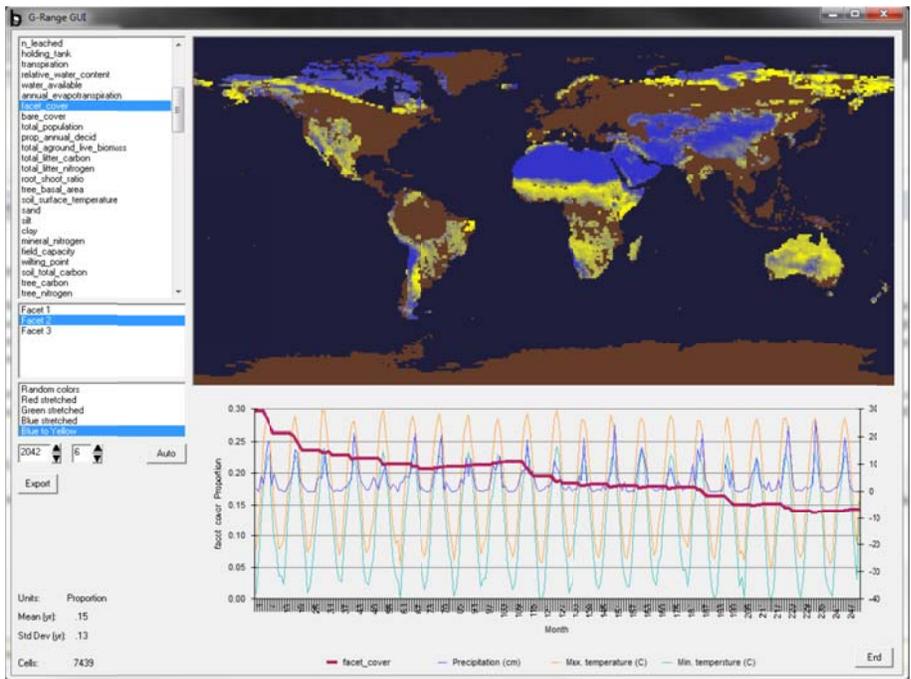
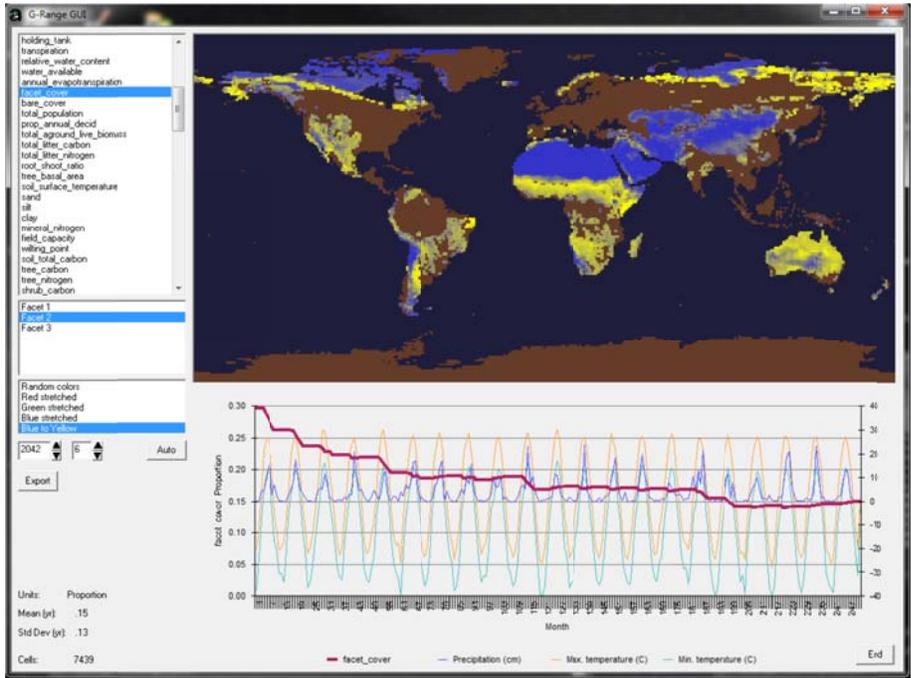
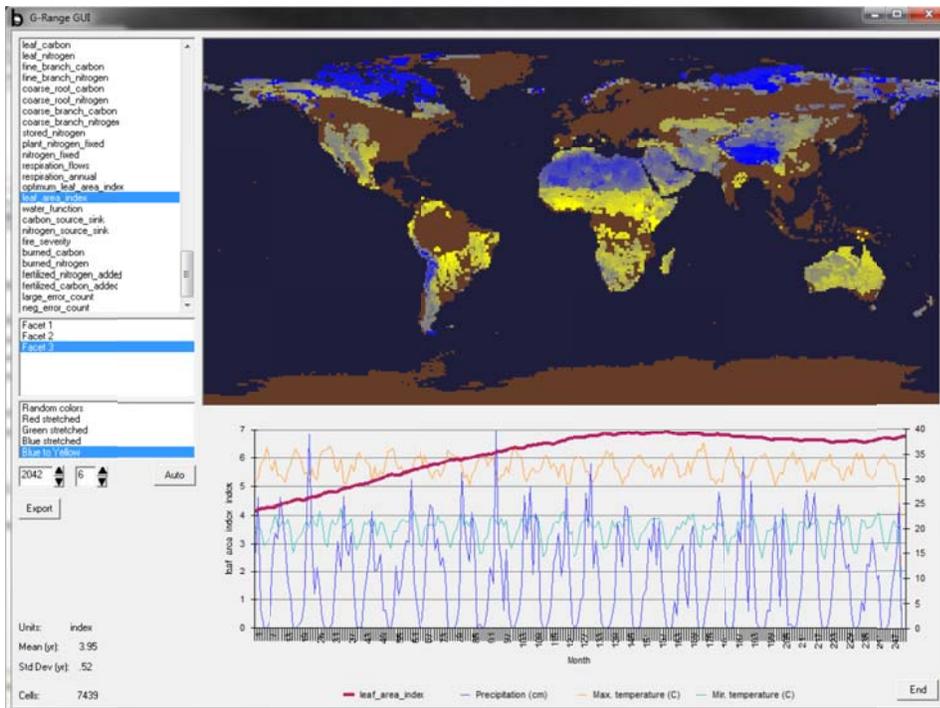
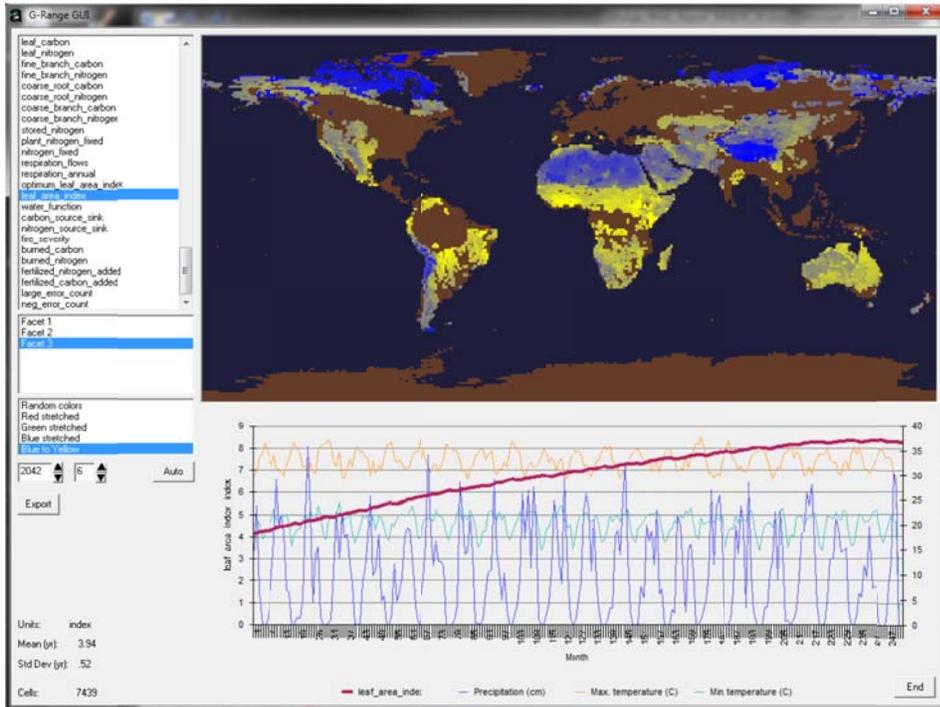


Figure 7. Comparisons of tree leaf area index in June 2042 under the A2 (a) and B1 (b) scenarios adopted in the IPCC (2010) reports. The graph is for a region in central Canada.



Appendix A. Simulation and state variables used in G-Range. Dimensions reflects the layers or dimensions of the attribute being represented (e.g., four soil layers or three vegetation facets). If Dimensions is blank, the variable represents one layer of information (e.g., a single daylength for any given cell).

Simulation variables	Meaning	Dimensions
x_dim	The x (east-west) dimension of the globe, in numbers of cells	
y_dim	The y (north-south) dimension of the globe, in numbers of cells	
lower_x	The lower x coordinate of maps (e.g., -180 degrees)	
lower_y	The lower y coordinate of maps (e.g., -90 degrees)	
upper_x	The upper x coordinate of maps (e.g., 180 degrees)	
upper_y	The upper y coordinate of maps (e.g., 90 degrees)	
cellsize	The dimension of landscape cells (e.g., 1 degree or 0.5 degree)	
range_cells	The number of range cells defined for the globe	
bin_path	The pathway to the executable G_Range.exe file	
app_path	The pathway to the G_Range application files	
parm_path	The pathway to the G_Range parameter files	
out_path	The pathway to the G_Range output folder	
zone_map	The file name of the zonal map	
land_map	The file name of a map distinguishing land and ocean	
elev_map	The file name of the elevation map	
class_map	The file name of the vegetation class map	
class_legend	The file name of legend to the class map, showing rangeland cells	
latitude_map	The file name of a map that identifies latitudes of cells	
landscape_type_map	The file name of the landscape unit map	
top_sand_map	The file name of a map showing topsoil percent sand	
top_silt_map	The file name of a map showing topsoil percent silt	
top_clay_map	The file name of a map showing topsoil percent clay	
top_gravel_map	The file name of a map showing topsoil percent gravel	
top_bulk_density_map	The file name of a map showing topsoil bulk density	
top_organic_carbon_map	The file name of a map showing topsoil organic carbon	
sub_sand_map	The file name of a map showing subsoil percent sand	
sub_silt_map	The file name of a map showing subsoil percent silt	

sub_clay_map	The file name of a map showing subsoil percent clay
sub_gravel_map	The file name of a map showing subsoil percent gravel
sub_bulk_density_map	The file name of a map showing subsoil bulk density
sub_organic_carbon_map	The file name of a map showing subsoil organic carbon
deciduous_tree_cover_map	The file name of a map of deciduous tree cover
evergreen_tree_cover_map	The file name of a map of evergreen tree cover
shrub_cover_map	The file name of a map of shrub percent cover
herb_cover_map	The file name of a map of herbaceous percent cover
precip_average_map	The file name of a map of average precipitation
temperature_average_map	The file name of a map of average temperature
precip_path_prefix	The path and prefix given to maps of precipitation
max_temp_path_prefix	The path and prefix given to maps of maximum temperature
min_temp_path_prefix	The path and prefix given to maps of minimum temperature
precip_temp_suffix	The suffix given to maps of precipitation and temperature
parms_file_name	The name of a file storing parameters of landscape units
start_yr	The starting year of the simulation
end_yr	The ending year of the simulation
state_var_flag	A switch showing whether or not state variables should be saved
state_var_file_out	The file name of the state variable file to be written
state_var_file_in	The file name of the state variable file to be read

Rangeland cell variables	Meaning	Dimensions
Zone	The zonal identifier of the landscape cell	
X	The x position of the cell, in integer grid units	
Y	The y position of the cell, in integer grid units	
range_type	The type of rangeland	
last_month_day_length	The day length of the last month simulated	
day_length_increasing	A switch showing if day length is increasing since last month	
day_length	The day length	
heat_accumulation	Heat accumulation	
facet_cover	The proportion cover of the facet	Facets
total_population	The population of plants for the vegetation layer	Veg. layers

bare_cover	The proportion of bare ground	
prop_annual_decid	The proportion of annual herbs or deciduous woody plants	Facets
pot_evap	Potential evapotranspiration	
snow	Snow accumulation	
snow_liquid	Snow accumulation, in liquid equivalents	
melt	Snow that has melted	
pet_remaining	Potential evapotranspiration remaining after some allocation	
ppt_soil	Precipitation in soil and available to plants	
runoff	Runoff	
ratio_water_pet	The ratio of available water to evapotranspiration	
co2_value	The concentration of carbon dioxide in the atmosphere	
pet_top_soil	Potential evapotranspiration from the top soil	
n_leached	Nitrogen that is leached	Soil layers
asmos	A holding tank variable, used to sum water availability	Soil layers
amov	A variable used to sum water movement	Soil layers
storm_flow	Storm flow	
holding_tank	Temporary water storage	
transpiration	Transpiration from plants	
relative_water_content	Relative water content	Soil layers
water_available	Water available for plant growth	Soil layers, except bottom
total_aground_live_biomass	Total aboveground live biomass	Facets
total_litter_carbon	Total litter carbon	Surface and soil
total_litter_nitrogen	Total litter nitrogen	Surface and soil
root_shoot_ratio	Root to shoot ratio	Facets
tree_basal_area	Tree basal area	
soil_surface_temperature	Soil surface temperature	
sand	Proportion sand	Soil layers
silt	Proportion silt	Soil layers
clay	Proportion clay	Soil layers
mineral_nitrogen	Mineral nitrogen	Soil layers
soil_depth	Soil depth	Soil layers

field_capacity	Soil field capacity	Soil layers
wilting_point	Soil wilting point	Soil layers
soil_total_carbon	Total soil carbon	
carbon_nitrogen_ratio	Carbon to nitrogen ratio	Surface and soil
fast_soil_carbon	Soil carbon with rapid turnover	Surface and soil
intermediate_soil_carbon	Soil carbon with intermediate turnover	
passive_soil_carbon	Soil carbon with very slow turnover	
fast_soil_nitrogen	Soil nitrogen with rapid turnover	Surface and soil
intermediate_soil_nitrogen	Soil nitrogen with intermediate turnover	
passive_soil_nitrogen	Soil nitrogen with very slow turnover	
potential_production	Potential product of all vegetation types	
belowground_pot_production	Belowground potential production	Veg. layers
aboveground_pot_production	Aboveground potential production	Veg. layers
total_pot_production	Total potential production	Veg. layers
co2_effect_on_production	Carbon dioxide concentration effect on production	
total_pot_prod_limited_by_n	Total potential production as limited by nitrogen	Veg. layers
fraction_live_removed_grazing	The fraction of live biomass removed by grazing	
fraction_dead_removed_grazing	The fraction of dead biomass removed by grazing	
temp_effect_on_decomp	The effect of temperature on decomposition	
water_effect_on_decomp	The effect of water availability on decomposition	
anerobic_effect_on_decomp	The effect of anerobic conditions on decomposition	
all_effects_on_decomp	Combined effects on decomposition	
dead_standing_carbon	Dead standing carbon	
dead_standing_nitrogen	Dead standing nitrogen	Facets
lignin_fine_root	Lignin in fine roots	Facets
lignin_coarse_root	Lignin in coarse roots	Facets
lignin_fine_branch	Lignin in fine branches	Facets
lignin_coarse_branch	Lignin in coarse branches	Facets
lignin_leaf	Lignin in leaves	Facets
litter_structural_carbon	Structural carbon in litter	Surface and soil
litter_metabolic_carbon	Metabolic carbon in litter	Surface and soil

litter_structural_nitrogen	Structural nitrogen in litter	Surface and soil
litter_metabolic_nitrogen	Metabolic nitrogen in litter	Surface and soil
tnetmin	Temporary storage of nitrogen accumulator	Surface and soil
tminup	Temporary storage of nitrogen accumulator	Surface and soil
grossmin	Gross mineralization of nitrogen	Surface and soil
volitn	Temporary storage of volitolized nitrogen	Surface and soil
fixnit	Nitrogen that is fixed	
runoffn	Nitrogen lost through runoff	
e_up	Nitrogen uptake	Facets and Woody parts
volatized_n	Volatized nitrogen	
phenology	Phenology of plants	Facets
fine_root_carbon	Carbon in fine roots	Facets
fine_root_nitrogen	Nitrogen in fine roots	Facets
seed_carbon	Carbon in seeds	Facets
seed_nitrogen	Nitrogen in seeds	Facets
leaf_carbon	Carbon in leaves	Facets
leaf_nitrogen	Nitrogen in leaves	Facets
fine_branch_carbon	Carbon in fine branches	Facets
fine_branch_nitrogen	Nitrogen in fine branches	Facets
coarse_root_carbon	Carbon in coarse roots	Facets
coarse_root_nitrogen	Nitrogen in coarse roots	Facets
coarse_branch_carbon	Carbon in coarse branches	Facets
coarse_branch_nitrogen	Nitrogen in coarse branches	Facets
stored_nitrogen	Stored nitrogen	Facets
plant_nitrogen_fixed	Nitrogen from plants that is fixed	Facets
respiration_flows	Respiration flows	Facets
carbon_source_sink	The source and sink for carbon	
nitrogen_source_sink	The source and sink for nitrogen	
optimum_leaf_area_index	Optimum leaf area indices of plants	Facets
leaf_area_index	Leaf area index	Facets
water_function	Water function influencing leaf and shoot death rate	

Appendix B. The main G-Range module, which calls submodules. Detailed scheduling of G-Range may be inferred from this module.

```
program GRange
```

```
! G-Range is a global model that simulates plant and animal dynamics on rangelands.
```

```
! The following uses calls per rangeland cell. That may be less than efficient, but most easily parallizable.
```

```
!
```

```
! R.B. Boone and R.T. Conant Last modified: March 5, 2011
```

```
use Structures
```

```
use Parameter_Vars
```

```
implicit none
```

```
integer icell
```

```
call Initialize_Parms ! Read the initial parameters for the simulation.
```

```
call Initialize_Landscape_Parms ! Read the parameters that correspond to the different landscape units simulated.
```

```
call Initialize_Globe ! Do the spatial initialization, setting up the global data.
```

```
call Initialize_Rangelands ! Do additional initialization of rangelands, including checking state variable file.
```

```
call Initialize_Outputs ! Produces positional data for rangelands.
```

```
do year = Sim_Parm%start_yr, Sim_Parm%end_Yr ! Loop through each year ...
```

```
call Each_Year ! Miscellaneous steps that need to be done each year
```

```
do month = 1, 12 ! ... and each month. Simulations must end in December
```

```
call Progress ! Report the progress of the simulation
```

```
call Read_Weather ! Get the month's weather data
```

```
call Read_Other ! Get any maps associated with fire or management
```

```
do icell = 1, range_cells ! Process all of the cells classed as rangeland only
```

```
call Update_Vegetation (icell) ! Update metrics for vegetation.
```

```

call Update_Weather (icell)           ! Calculate snowfall, evapotranspiration, etc. Also updates heat accumulation.
call Potential_Production (icell)    ! Calculate potential production, and plant allometrics adjusted by grazing fraction
call Herb_Growth (icell)            ! Calculate herbaceous growth
call Woody_Growth (icell)          ! Calculate woody plant growth
call Grazing (icell)                ! Remove material grazed by livestock
call Plant_Part_Death (icell)      ! Plant part death
call Whole_Plant_Death (icell)     ! Whole plant death
call Management (icell)          ! Fertilization and other management
call Plant_Reproduction (icell)   ! Seed-based reproduction by plants
call Update_Vegetation (icell)    ! Update metrics for vegetation
call Water_Loss (icell)           ! Calculate water loss
call Decomposition (icell)       ! Decomposition
call Nitrogen_Losses (icell)     ! Leaching and volatilization of nitrogen
end do

call Update_Vegetation             ! Update metrics for vegetation
call Each_Month                   ! Miscellaneous steps that need to be done each month
call Output_Surfaces              ! Produce output surfaces
end do
! End monthly loop
end do
! End yearly loop

call Wrap_Up                       ! Do simulation-ending tasks, including some output
write(*,*) ' '
write(*,*) 'G-Range is complete.'

end program

```

Appendix C. Parameters within the landscape unit parameter file (e.g., Land_Units.grg in the base application). The parameters shown are for rangeland cells within a single land cover type. Analogous blocks of parameters would follow those shown for the remaining land cover types in the spatial layer used.

```

10          // range_type  GRASSLAND / STEPPE
              The landscape unit identifier
8.00        // prcp_threshold
              The amount of precipitation required for there to be runoff (in cm)
0.15        // prcp_threshold_fraction
              The fraction of monthly precipitation that is lost as storm runoff (unitless)
0.20        // base_flow_fraction
              The fraction of soil water content in the last soil layer lost to base flow (unitless)
0.8, 0.6, 0.4, 0.2 // soil_transpiration_fraction
              The fraction of water transpired from each soil layer (unitless)
22.0        // init_soil_c_n_ratio
              Initial soil carbon to nitrogen ratio
150.0       // init_lignin_n_ratio
              Initial lignin to nitrogen ratio in litter
1200, 500, 800, 3500, 1200, 100, 100, 300, 200, 300 // tree_carbon
              Initial tree carbon for each plant part (1-5 = leaf, fine root, fine branch, coarse branch, coarse root) for alive (1)
              and dead (2) components (gC m^2)
300, 150, 200, 800, 300, 40, 40, 80, 60, 80 // shrub_carbon
              Initial shrub carbon for each plant part (1-5 = leaf, fine root, fine branch, coarse branch, coarse root) for alive (1)
              and dead (2) components (gC m^2)
0.5, 2.0, 8.0 // plant_dimension
              Single dimension of square area occupied by plant root mass per facet (1-3 = herb, shrub, tree) (m)
30., 45., 1.0, 2.5 // temperature_production
              Effect of temperature on potential production, coefficients shaping a response curve, 1 = optimum temperature,
              2 = maximum temperature, 3 = left shape, 4 = right shape (C, C, unitless, unitless)
60.0        // standing_dead_production_halved
              Standing dead material plus a portion of structural material that reduces production by half due to physical

```

obstruction (gC m²)
 0.40 // radiation_production_coefficient
 Coefficient relating potential aboveground monthly production as a function of solar radiation outside the atmosphere (unitless)

0.40, 0.37, 0.33 // fraction_carbon_to_roots
 Fraction of carbon allocated to roots (unitless)

1 // grazing_effect
 Grazing effect flag, used to specify 0 through 6 types of plant functional responses to grazing. See documentation for their definitions.

0.8 // effect_of_co2_on_transpiration
 The effect of CO₂ on transpiration rates (unitless)

2.0, 2.0 // decomp_rate_structural_litter_inverts
 Decomposition rate of structural litter in soil layers one and two, due to invertebrates (gC m²)

0.25 // feces_lignin
 Proportion of feces that is lignin (unitless)

0.0200, 0.0012, 0.2600, -0.0015 // lignin_content_fraction_and_precip
 Relating precipitation to lignin content in materials, providing a 1) intercept aboveground, 2) slope aboveground, 3) intercept belowground, 4) slope belowground

0.2 // fraction_urine_volatized
 Fraction of urine nitrogen that is volatized (unitless)

0.05000, 0.00700 // precip_n_deposition
 Coefficients shaping a line relating precipitation to nitrogen deposition (intercept and slope)

30.00, 0.0100 // precip_n_symbiotic
 Coefficients shaping a line relating precipitation to symbiotic nitrogen fixation (intercept and slope)

0.25 // decomp_litter_mix_facets
 The degree to which litter mixes between facets within a landscape cell (proportion, from 0 to 1)

0,0, 200,1, 400,2, 700,3, 1200,4, 0,0, 400,1, 800,2, 1000,3, 2000,4, 0,0, 400,1, 800,2, 1000,3, 2000,4 // degree_days_phen
 Degree days relating to plant phenology, per facet, and with four pairs of values forming the relationship (degree days per phenology, which is from 0 to 4)

1500, 3000, 3000 // degree_days_reset
 Degree days to reset phenology back to zero (degree days)

1.0 // tree_site_potential

Tree site potential, in peak aboveground herbaceous biomass if trees are absent (gB m²)

0.001 // max_symbiotic_n_fixation_ratio
Maximum symbiotic nitrogen fixation, in gN fixed for gC of new growth

10., 13., 0., 0., 0., 13., 20., 30., 50., 60., 15., 21., 32., 52. 52. // minimum_c_n_ratio
Minimum carbon to nitrogen ratio of plant biomass, for three facets, five plant parts each (the last three in herbs not used) (unitless)

30., 33., 0., 0., 0., 33., 40., 50., 80., 90., 35., 51., 62., 92. 95. // maximum_c_n_ratio
Maximum carbon to nitrogen ratio of plant biomass, for three facets, five plant parts each (the last three in herbs not used) (unitless)

6.0 // maximum_leaf_area_index
Maximum leaf area index (unitless)

2000. // k_leaf_area_index
Large wood mass at which half of the maximum leaf area index is attained (gC m²)

0.008 // biomass_to_leaf_area_index_factor
Biomass to leaf area index conversion factor (coefficient)

0.020 // annual_fraction_volatilized_n
Annual fraction of nitrogen that is volatilized (unitless)

0.10, 0.10, 0.10 // maximum_root_death_rate
Maximum root death rate, per facet (herb, shrub, and tree) (unitless)

0.20, 0.95, 0.20, 150.0 // shoot_death_rate
Shoot death rate due to 1) water stress, 2) phenology, 3) shading according to the carbon concentration in 4.

0.20 // prop_annuals
Proportion of herbaceous plants that are annuals (unitless)

12 // month_to_remove_annuals
Month to kill annual plants if they had not been killed already by phenology or other reasons (month)

1050.0, 1050.0, 1050.0 // relative_seed_production
Relative seed production of plants in each facet (herbs, shrubs, trees) (unitless)

0.67,0.2, 2.9,1.0, 3.0,0.2, 6.0,1.0, 0.2,0.2, 0.4,1.0 // water_effect_on_establish
The effect of available water ratio to potential evapotranspiration on plant establishment, per facet (herb, shrub, tree) with two pairs of values defining a linear relationship

100.0,1.0, 600.,0.3, 0.0,1.0, 300.0,0.5, 150.0,1.0, 300.0,0.4 // herb_root_effect_on_establish
The effect of herbaceous root biomass on the establishment of plants, per facet (herb, shrub, tree)

with two pairs of values defining a linear relationship
 300.0,1.0, 1000.0,0.0, 300.0,1.0, 1000.0,0.0, 300.0,1.0, 1000.0,0.0 // litter_effect_on_establish
 The effect of surface litter biomass on the establishment of plants, per facet (herb, shrub, tree) with
 two pairs of values defining a linear relationship
 0.0,1.0, 0.4,0.1, 0.0,1.0, 0.8,0.1, 0.0,1.0, 0.8,0.1 // woody_cover_effect_on_understory
 The effect of woody cover on the establishment of plants in understories, per facet (herb, shrub, tree)
 with two pairs of values defining a linear relationship
 0.05, 0.03, 0.01 // nominal_plant_death_rate
 Nominal plant death rate, per facet (unitless)
 2.0,0.0, 0.5,0.6, 1.0,0.0, 0.3,0.3, 0.9,0.0, 0.2,0.3 // water_effect_on_death_rate
 The effect of available water ratio to potential evapotranspiration on plant death rate, per facet
 (herb, shrub, tree) with two pairs of values defining a linear relationship
 0.0,0.0, 1.0,0.5, 0.0,0.0, 1.0,0.1, 0.0,0.0, 1.0,0.1 // grazing_effect_on_death_rate
 The effect of grazing frequency on plant death rate, per facet (herb, shrub, tree) with two pairs of values
 defining a linear relationship
 0.0,0.0, 5.0,0.05, 0.0,0.0, 4.0,0.05, 0.0,0.0, 4.0,0.01 // shading_effect_on_death_rate
 The effect of shading on plant death rate, per facet (herb, shrub, tree) with two pairs of values defining
 a linear relationship
 0.02, 0.05, 0.05 // fall_rate_of_standing_dead
 The rate standing dead falls to join litter, per facet (herb, shrub, tree) (unitless)
 0.10 // death_rate_of_deciduous_leaves
 The rate of death of deciduous leaves during the period of senescence (unitless)
 0.2, 0.2, 0.2 // drought_deciduous
 The fraction of plants that are drought deciduous, per facet (herb, shrub, tree) (unitless)
 0.3 // fraction_woody_leaf_n_translocated
 The fraction of nitrogen in woody leaves that are translocated back to roots prior to death (unitless)
 0.08, 0.03, 0.01 // leaf_death_rate
 Leave death rate, per facet (herb, shrub, tree) (unitless)
 0.03, 0.02, 0.02 // fine_root_death_rate
 Fine root death rate, per facet (herb, shrub, tree) (unitless)
 0.0, 0.005, 0.005 // fine_branch_death_rate
 Fine branch death rate, per facet (herb, shrub, tree) (unitless)

0.0, 0.003, 0.003 // coarse_branch_death_rate
Coarse branch death rate, per facet (herb, shrub, tree) (unitless)

0.0, 0.005, 0.005 // coarse_root_death_rate
Coarse root death rate, per facet (herb, shrub, tree) (unitless)

0.3 // fraction_carbon_grazed_returned
The fraction of carbon that is grazed that is returned through feces or other routes (unitless)

0.5 // fraction_excreted_nitrogen_in_feces
The fraction of excrete nitrogen that is feces; the remainder is urine (unitless)

0.8, 0.15, 0.05 // fraction_grazed_by_facet
The fraction of total grazing that is from each of the facets (herb, shrub, tree), with the total summing to 1.0 (proportion)

0.35 // fraction_grazed
The annual proportion of plant material grazed

0.0 // frequency_of_fire
The probability of fire per year for any given cell within the landscape unit
(NOTE SCALE DEPENDENCE, USE DEPENDS ON fire_maps_used), set to 0 for no fire (unitless)

0.0 // fraction_burned
The proportion of a landscape cell that burns, in the case of a fire event
(NOTE SCALE DEPENDENCE. ALSO ONE FIRE PER YEAR MAX) (unitless)

6 // burn_month
The month in which patches will be burned, in the case of a fire event (ONE FIRE PER YEAR MAX, USE DEPENDS ON fire_maps_used) (month)

50., 400. // fuel_vs_intensity
The fuel load as related to low and high intensity fires (g biomass / m²)

0., 1., 0.3, 0.7 // green_vs_intensity
The proportion of aboveground vegetation that is green versus fire intensity (unitless)

0.1,1.0, 0.1,0.2, 0.1,0.2 // fraction_shoots_burned
The proportion of live leaves and shoots removed by a fire event, by facet, for low and high intensity fire (unitless)

0.4,1.0, 0.3,0.9, 0.3,0.9 // fraction_standing_dead_burned
The proportion of standing dead removed by a fire event, by facet, for low and high intensity fire (unitless)

0.2,0.5, 0.0,0.15, 0.0,0.15 // fraction_plants_burned_dead

0.1, 0.5, 0.1, 0.5, 0.1, 0.5 The proportion of plants that are burned that die, by facet, for low and high intensity fire (unitless)
 // fraction_litter_burned
 0.06 The proportion of litter removed by a fire event, by facet, for low and high intensity fire (unitless)
 // fraction_burned_carbon_as_ash
 0.08 The proportion of carbon in burned aboveground material that is ash, going to structural litter (unitless)
 // fraction_burned_nitrogen_as_ash
 0.0 The proportion of nitrogen in burned aboveground material that is ash, going to soil mineral nitrogen (unitless)
 // frequency_of_fertilization
 0.0001 The probability of fertilization per year in the landscape unit (USE DEPENDS ON fertilize_maps_used) (unitless)
 // fraction_fertilized
 The proportion of a landscape cell that is fertilized, in the case of a fertilization event
 (NOTE SCALE DEPENDENCE. USE DEPENDS ON fertilize_maps_used) (unitless)
 6 // fertilize_month
 The month in which fertilization occurs (one event per year per landscape unit) (month)
 3.0 // fertilize_nitrogen_added
 Amount of inorganic nitrogen added during a fertilization event (g / m2)
 100.0 // fertilize_carbon_added
 Amount of carbon added as part of organic matter fertilizer (g / m2)

Appendix D. The spatial layers used in the base application of G-Range, with 1 degree resolution surfaces used in examples. Other names have a form similar to *0p5, *0p25, *0p167, *0p1, and *0p083. Additional spatial surfaces controlling fire and fertilization are included, but not used in the base application.

Spatial layer name	Contents	Units	Range
decid_1p0.asc	Deciduous tree cover ^a	Percent	0 – 100
egreen_1p0.asc	Evergreen tree cover ^a	Percent	0 – 100
goge1p0.asc	Global Land Cover Characterization ^b	Type	1 – 100
herbcv_1p0.asc	Herbaceous cover ^c	Percent	0 – 100
land1p0.asc	Land versus ocean ^d	Switch	0, 1
lats1p0.asc	Latitude of cell center ^d	Degree	-90 – 90
prcp_avg_1p0.asc	Average total annual precipitation ^e	Millimeters	0 – max
sage1p0.asc	Landscape units (potential vegetation) ^f	Type	1 – 15
shrbcv_1p0.asc	Shrub cover ^g	Percent	0 – 100
sub_bulk1p0.asc	Subsoil bulk density ^h	g cm ⁻³	0 – max
sub_carbon1p0.asc	Subsoil organic carbon ^h	Percent weight	0 – 100
sub_clay1p0.asc	Subsoil clay ^h	Percent weight	0 – 100
sub_gravel1p0.asc	Subsoil gravel ^h	Percent volume	0 – 100
sub_sand1p0.asc	Subsoil sand ^h	Percent weight	0 – 100
sub_silt1p0.asc	Subsoil silt ^h	Percent weight	0 – 100
temp_avg_1p0.asc	Average annual temperature ^e	Degree C	min - max
top_bulk1p0.asc	Topsoil bulk density ^h	g cm ⁻³	0 – max
top_carbon1p0.asc	Topsoil organic carbon ^h	Percent weight	0 – 100
top_clay1p0.asc	Topsoil clay ^h	Percent weight	0 – 100
top_gravel1p0.asc	Topsoil gravel ^h	Percent volume	0 – 100
top_sand1p0.asc	Topsoil sand ^h	Percent weight	0 – 100
top_silt1p0.asc	Topsoil silt ^h	Percent weight	0 – 100
treecv_1p0.asc	Tree cover ^c	Percent weight	0 – 100
zone1p0.asc	Cell zonal identifier ^d	Identifier	1 – max
p????+_1p0.asc	Precipitation, yr (????), mn (++) ^e	Millimeters	0 – max
x????+_1p0.asc	Maximum temperature, yr (????), mn (++) ^e	Degree C	min – max
n????+_1p0.asc	Minimum temperature, yr (????), mn (++) ^e	Degree C	min – max

- a – DeFries (2000), describing the spatial layers available from the online source: Tree Cover Continuous Fields. Global Land Cover Facility, <http://glcf.umiacs.umd.edu/data/treecover/>
- b – Global Land Cover Characterization, Version 2. Available from the online source: Global Land Cover Characterization, US Department of the Interior, US Geological Survey, <http://edc2.usgs.gov/glcc/glcc.php>
- c – MODIS Vegetation Continuous Fields (Hansen et al. 2006), e.g., <http://glcf.umd.edu/data/vcf/>
- d – Computed from geographic principles, widely available spatial data (e.g., global countries), or custom developed.
- e – CRU TS3.0 precipitation, maximum, and minimum temperature, or derived products (CRU 2008), <http://badc.nrec.ac.uk/browse/badc/cru/>
- f – Global Potential Vegetation Dataset. Ramankutty and Foley (1999), and available from http://www.sage.wisc.edu/download/potveg/global_potveg.html
- g – Loosely derived from MODIS Vegetation Continuous Field (Hansen et al. 2006), e.g., <http://glcf.umd.edu/data/vcf/>. A high quality shrub cover dataset does not exist for the globe.
- h – Harmonized World Soil Database, FAO/IIASA/ISRIC/ISSCAS/JRC, 2009. Harmonized World Soil Database (version 1.1). FAO, Rome, Italy and IIASA, Laxenburg, Austria. <http://www.iiasa.ac.at/Research/LUC/External-World-soil-database/HTML/>

Appendix E. Main output files produced by G-Range. These are binary files with a header preceding each spatial surface composed of the number of rangeland cells, the year, and the month of the simulation. That information is followed by a number of four-byte floating point values equal to the number of rangeland cells. If layers is blank, the month is represented by a single spatial layer.

File name	Layers	Units	Description
x_loc		coordinate	X dimension of rangeland cell
y_loc		coordinate	Y dimension of rangeland cell
range_type		category	Identifier storing the type of rangeland cell; its landscape unit
precip		cm/month	Precipitation, from the global database
max_temp		degrees C	Maximum temperature, from the global database
min_temp		degrees C	Minimum temperature, from the global database
day_length		hours/day	Day length, calculated based on latitude and month
heat_accumulation		degree C days	Heat accumulation, above a threshold of 4.4 C
pot_evap		cm/month	Potential evapotranspiration for the cell
evaporation		cm/month	Water evaporated from the soil and vegetation
snow		Cm	Snowpack
snow_liquid		Cm	Snowpack liquid water.
melt		Cm	Snow that melts from snowpack
pet_remaining		Cm	Potential evaporation decremented as steps are calculated
ppt_soil		Cm	Precipitation adjusted for snow accumulation and melt
runoff		Cm	Runoff from the rangeland cell
ratio_water_pet		Ratio	Ratio of available water to potential evapotranspiration
co2_value		coefficient	CO2 effect on evapotranspiration
pet_top_soil		cm/day	Potential evaporation from top soil
n_leached	Soil layer	g/m ² /layer	Nitrogen leached from soil
holding_tank		cm/layer	Stores water temporarily
transpiration		Cm	Transpiration water loss
relative_water_content	Soil layer	Cm	Used to initialize
water_available	Entry	Cm	Water available for growth (1), survival (2), and top layers (3)
annual_evapotranspiration		Cm	Annual actual evapotranspiration

facet_cover	Facet	Proportion	The proportion occupied by each facet
bare_cover		Proportion	Bare cover stored, rather than continually summing the three facets
total_population	Veg layer	Population	The total population of each vegetation layer
prop_annual_decid	Facet	Proportion	Proportion of facet that is annual plants or deciduous
total_aground_live_biomass	Facet	g/ m ²	Total aboveground green biomass
total_litter_carbon	Layer	g/ m ²	Average monthly litter carbon, for surface and soil
total_litter_nitrogen	Layer	g/ m ²	Average monthly litter carbon, for surface and soil
root_shoot_ratio	Facet	Ratio	Root shoot ratio
tree_basal_area		m ²	Basal area for trees
soil_surface_temperature		C	Average soil surface temperature
sand	Soil layer	Proportion	The percent sand in the soil
silt	Soil layer	Proportion	The percent silt in the soil
clay	Soil layer	Proportion	The percent clay in the soil
mineral_nitrogen	Soil layer	g/m ²	Mineral nitrogen content for layer
field_capacity	Soil layer	cm/layer	Field capacity for the four soils layers
wilting_point	Soil layer	cm/layer	Wilting point for the four soil layers
soil_total_carbon		g/ m ²	Total soil carbon
tree_carbon	Wood part	g/ m ²	Tree carbon in its components
tree_nitrogen	Wood part	g/ m ²	Tree nitrogen in its components
shrub_carbon	Wood part	g/ m ²	Shrub carbon in its components
shrub_nitrogen	Wood part	g/ m ²	Shrub nitrogen in its components
carbon_nitrogen_ratio	Layer	Ratio	Carbon to nitrogen ratio
fast_soil_carbon	Layer	g/ m ²	Soil organic matter carbon
intermediate_soil_carbon		g/ m ²	Intermediate soil carbon
passive_soil_carbon		g/ m ²	Passive soil carbon
fast_soil_nitrogen	Layer	g/ m ²	Soil organic matter nitrogen
intermediate_soil_nitrogen		g/ m ²	Intermediate soil nitrogen
passive_soil_nitrogen		g/ m ²	Passive soil nitrogen
potential_production		coefficient	Calculated potential production for the cell, as in index
belowground_pot_production	Veg layer	g/ m ²	Biomass, belowground potential production
aboveground_pot_production	Veg layer	g/ m ²	Biomass, aboveground potential production

total_pot_production	Veg layer	g/ m ²	Biomass, calculate total potential production
co2_effect_on_production		coefficient	Effect of CO ₂ increasing from 350 to 700 ppm on production
total_pot_prod_limited_by_n	Veg layer	coefficient	Total potential production with limits due to nitrogen
fraction_live_removed_grazing		proportion	Fraction of live forage removed by grazing
fraction_dead_removed_grazing		proportion	Fraction of dead forage removed by grazing
temp_effect_on_decomp		coefficient	Temperature effect on decomposition
water_effect_on_decomp		coefficient	Water effect on decomposition
anerobic_effect_on_decomp		coefficient	Anerobic effects on decomposition
all_effects_on_decomp		coefficient	Combined effects on decomposition
dead_fine_root_carbon	Facet	g/ m ²	Dead fine root carbon
dead_fine_root_nitrogen	Facet	g/ m ²	Dead fine root nitrogen
dead_standing_carbon	Facet	g/ m ²	Standing dead carbon of leaf and stem
dead_standing_nitrogen	Facet	g/ m ²	Standing dead nitrogen of leaf and stem
dead_seed_carbon	Facet	g/ m ²	Dead seed carbon
dead_seed_nitrogen	Facet	g/ m ²	Dead seed nitrogen
dead_leaf_carbon	Facet	g/ m ²	Dead leaf carbon
dead_leaf_nitrogen	Facet	g/ m ²	Dead leaf nitrogen
dead_fine_branch_carbon	Facet	g/ m ²	Dead fine branch carbon
dead_total_fine_branch_carbon		g/ m ²	Dead fine branch carbon, summed across facets
dead_fine_branch_nitrogen	Facet	g/ m ²	Dead fine branch nitrogen
dead_total_fine_branch_nitrogen		g/ m ²	Dead fine branch nitrogen, summed across facets
dead_coarse_root_carbon	Facet	g/ m ²	Dead coarse root carbon
dead_total_coarse_root_carbon		g/ m ²	Dead total coarse root carbon, summed across facets
dead_coarse_root_nitrogen	Facet	g/ m ²	Dead coarse root nitrogen
dead_total_coarse_root_nitrogen		g/ m ²	Dead total coarse root nitrogen, summed across facets
dead_coarse_branch_carbon	Facet	g/ m ²	Dead coarse branch carbon
dead_total_coarse_branch_carbon		g/ m ²	Dead total coarse branch carbon, summed across facets
dead_coarse_branch_nitrogen	Facet	g/ m ²	Dead coarse branch nitrogen
dead_total_coarse_branch_nitrogen		g/ m ²	Dead total coarse branch nitrogen, summed across facets
lignin_fine_root	Facet	g/ m ²	Fine root lignin concentration
lignin_coarse_root	Facet	g/ m ²	Coarse root lignin concentration

lignin_fine_branch	Facet	g/ m ²	Fine branch lignin concentration
lignin_coarse_branch	Facet	g/ m ²	Coarse branch lignin concentration
lignin_leaf	Facet	g/ m ²	Leaf lignin concentration
litter_structural_carbon	Layer	g/ m ²	Litter structural carbon at the surface and in the soil
litter_metabolic_carbon	Layer	g/ m ²	Litter metabolic carbon at the surface and in the soil
litter_structural_nitrogen	Layer	g/ m ²	Litter structural nitrogen at the surface and in the soil
litter_metabolic_nitrogen	Layer	g/ m ²	Litter structural nitrogen at the surface and in the soil
maintain_respiration	Facet	g/ m ²	Maintenance respiration
Phenology	Facet	g/ m ²	Phenological stage
fine_root_carbon	Facet	g/ m ²	Fine root carbon
fine_root_nitrogen	Facet	g/ m ²	Fine root nitrogen
seed_carbon	Facet	g/ m ²	Seed carbon
seed_nitrogen	Facet	g/ m ²	Seed nitrogen
leaf_carbon	Facet	g/ m ²	Leaf carbon
leaf_nitrogen	Facet	g/ m ²	Leaf nitrogen
fine_branch_carbon	Facet	g/ m ²	Fine branch carbon
fine_branch_nitrogen	Facet	g/ m ²	Fine branch nitrogen
coarse_root_carbon	Facet	g/ m ²	Coarse root carbon
coarse_root_nitrogen	Facet	g/ m ²	Coarse root nitrogen
coarse_branch_carbon	Facet	g/ m ²	Coarse branch carbon
coarse_branch_nitrogen	Facet	g/ m ²	Coarse branch nitrogen
stored_nitrogen	Facet	g/ m ²	Stored nitrogen
plant_nitrogen_fixed	Facet	g/ m ²	Plant nitrogen fixed
nitrogen_fixed	Facet	g/ m ²	Nitrogen fixed
respiration_flows	Facet	g/ m ²	Maintenance respiration flows to storage pool
respiration_annual	Facet	g/ m ²	Maintenance respiration flows for year
optimum_leaf_area_index	Facet	index	Optimum leaf area index
leaf_area_index	Facet	index	Leaf area index
water_function		coefficient	Water function influencing mortality
carbon_source_sink		g/ m ²	Carbon pool
nitrogen_source_sink		g/ m ²	Nitrogen pool

fire_intensity	Index	Fire intensity
burned_carbon	g/ m ²	Burned carbon, excluding whole plant death
burned_nitrogen	g/ m ²	Burned nitrogen, excluding whole plant death
fertilized_nitrogen_added	g/ m ²	Fertilizer nitrogen added
fertilized_carbon_added	g/ m ²	Fertilizer carbon added
large_error_count	count	Large error count
neg_error_count	count	Negative error count

Appendix F. Details are provided for select subroutines in G-Range, with the notes more freeform than other portions of this report. Often looping structures are used in calculations (e.g., facets), but in the examples below, a single entry is used for brevity (e.g., Herbs). The focus is upon details that relate to ecosystem process modeling. Subroutines in the Beta version of G-Range include:

Initialization

Initialize_Parms

The subroutine is not involved in simulating ecosystem processes. It reads parameters that control general model execution. No details are of note.

Initialize_Globe

The subroutine is not involved in simulating ecosystem processes. The routine first reads the zonal map, which provides a unique identifier to every cell in the global spatial surface. The six-line header associated with that GRIDASCII file is processed, and the attributes of that map (i.e., the x dimension, y dimension, lower x, lower y, cellsize) are saved for use in the simulation (line six, the no data value, is not saved). An upper x and upper y value are calculated based on the lower x and lower y values plus the map dimensions. If the results exceed 180 and 90, the simulation will provide an error message and stop. In reading subsequent spatial layers, the first two lines of the header (x dimension and y dimension) are read and compared to the values from the zonal map. If they disagree, an error is given and the simulation tops. The remaining four header lines are ignored. The routine calls Read_Map, asking that a temporary array be filled with each spatial layer needed. The routine then transfers the map to its appropriate location in the Globe data structure.

Read_Map

The subroutine is not involved in simulating ecosystem processes. The file name to load and a map name are passed to this subroutine, then the spatial layer is read into a temporary array with the dimensions of the world.

Initialize_Landscape_Parms

The subroutine is not involved in simulating ecosystem processes. Parameters describing how ecosystem process in landscape units should be simulated are read. Landscape units are read in until an end of file marker is located. In the current version of G-Range, the routine is not robust to ill-formed parameter files. Note that selected parameters that appeared stable across parameter sets are hardwired in Initialize_Landscape_Parms.

Initialize_Rangelands

This subroutine alters the contents of rangeland cells, and so may have a direct outcome on the outcome of a simulation. The attributes of rangeland cells are initialized. First, the four soil layers are initialized. Our input soil information includes data for the topsoil and for the subsoil. Entries are weighted as follows, with the numbers indicated the soil layer and sand used as an example:

$$\text{Sand}(1) = \text{Top_Sand} * 1.0 + \text{Sub_Sand} * 0.0$$

$$\text{Sand}(2) = \text{Top_Sand} * 0.667 + \text{Sub_Sand} * 0.334$$

$$\text{Sand}(3) = \text{Top_Sand} * 0.334 + \text{Sub_Sand} * 0.667$$

$$\text{Sand}(4) = \text{Top_Sand} * 0.0 + \text{Sub_Sand} * 1.0$$

These values are then divided by 100 to yield proportions.

Potential (maximum) plant populations are calculated based on the dimensions of the plants provided in the landscape unit parameter file, e.g.:

$$\text{Indiv_Plant_Area}(\text{Herbs}) = \text{Plant_Dimen}(\text{Herbs}) * \text{Plant_Dimen}(\text{Herbs})$$

$$\text{Potential_Population}(\text{Herbs}) = 1 \text{ km}^2 / \text{Indiv_Plant_Area}(\text{Herbs})$$

Field capacity and wilting point are calculated based on values used in Century, which are based on Gupta and Larson (1979) and Rawls et al. (1982). For the first layer:

$$\text{Field_capacity}(1) = \text{Sand}(1) * 3.075 + \text{Silt}(1) * 5.866 + \text{Clay}(1) * 8.039 + \\ \text{Organic_carbon}(1) * 2.208 + \text{Bulk_density}(1) * -14.340$$

$$\text{Wilting_point}(1) = \text{Sand}(1) * -0.059 + \text{Silt}(1) * 1.142 + \text{Clay}(1) * 5.766 + \\ \text{Organic_carbon}(1) * 2.228 + \text{Bulk_density}(1) * 2.671$$

Field capacity and wilting point are then reduced to correct for gravel volume.

Relative water content is initialized to 0.5.

Total soil carbon is then initialized, using values cited in Century 4.5. Average silt and clay across all layers is used. Then soil total carbon is:

$$\text{Soil_total_carbon} = (-0.827 * \text{Avg_temperature} + 0.0024 * \\ \text{Avg_temperature}^2 + 0.127 * \text{Precip} + 0.000938 * \text{Precip}^2 +$$

$$0.0899 * \text{Precip} * \text{Silt} + 0.060 * \text{Clay} * \text{Precip} + 4.09) * 1000.0$$

The value is then set to 500 g / m² if it is less than that value.

Soil carbon is initialized into the fast, intermediate, and passive pools using values included in Century, which cites Burke's equations:

$$\text{Fast_soil_carbon}(\text{Surface}) = 10.0 + (10.0 * 0.011)$$

$$\text{Fast_soil_carbon}(\text{Soil}) = \text{Soil_total_carbon} * 0.02 + \\ (\text{Soil_total_carbon} * 0.02 * 0.011)$$

$$\text{Intermediate_soil_carbon} = \text{Soil_total_carbon} * 0.65 + \\ (\text{Soil_total_carbon} * 0.65 * 0.011)$$

$$\text{Passive_soil_carbon} = \text{Soil_total_carbon} * 0.34 + \\ (\text{Soil_total_carbon} * 0.34 * 0.011)$$

Soil nitrogen concentrations are calculated based on initialized soil carbon and the carbon nitrogen ratio provided in the parameter set.

Litter structural carbon is initialized to 100. Litter metabolic carbon is initialized to 100.0 * 0.011.

Carbon components for all facets are initialized following Century, as follows:

$$\text{Leaf_carbon}(\text{Herbs}) = 200 + (200 * 0.011)$$

$$\text{Leaf_nitrogen}(\text{Herbs}) = 3$$

$$\text{Dead_standing_carbon}(\text{Herbs}) = 80 + (80 * 0.011)$$

$$\text{Dead_standing_nitrogen}(\text{Herbs}) = 1.6$$

$$\text{Fine_root_carbon}(\text{Herbs}) = 200 + (200 * 0.011)$$

$$\text{Fine_root_nitrogen}(\text{Herbs}) = 3$$

Other carbon concentrations such as fine branch carbon and coarse root carbon are initialized to zero.

Herb, shrub, and tree facet covers are calculated based on spatial layer input, then checked to ensure they do not exceed 1.0. Proportions of annual and deciduous plants are calculated based on information in the spatial layers and landscape unit parameter file. Total populations for the different facets are then calculated, based on the facet cover and potential

population. Plants in the understory are calculated similarly, but then reduced by a coefficient (e.g., 0.1667 * Herbs to reduce herbs under trees), but these values are not critically defined; they set up initial values only.

Lastly, plant lignin fractions for both the surface and soil are initialized to 0.25.

Set_Defaults

The subroutine is not directly involved in simulating ecosystem processes, but may have an effect on simulation outcomes. For all cells in the rangeland program structure, select variables are initialized to 0.

Weather

Read_Weather

The subroutine is not involved in simulating ecosystem processes. It reads spatial layers reporting monthly precipitation, minimum temperature, and maximum temperature.

Update_Weather

The subroutine is not involved in simulating ecosystem processes, although storm flow and head accumulation are updated for the cell being processed. Mainly, **Update_Weather** calls the remaining weather-related subroutines, **P_Evap**, **Day_Length**, and **Snow_Dynamics**.

P_Evap

This subroutine calculates potential evapotranspiration for a landscape cell using relationships first proposed by Penman and modified by Monteith (Penman 1948; Monteith 1965). These are long-established relationships are not further detailed here.

Day_Length

This subroutine uses trigonometric transformations and coefficients to calculate day length for any landscape cell, given its latitude in radians and the day-of-the-year for the middle of the month in question. These transformations are well established and won't be further detailed here. The **Day_Length** subroutine compares the day length calculated for the cell to the day length stored from the previous month. If the day length is increasing, heat accumulation is reset to zero, and a flag showing an increasing day length is set.

Shortwave (function)

This function uses trigonometric transformations and the day-of-the-year of the middle of the month to calculate solar radiation for a cell, given the latitude of the cell in radians. Solar radiation is calculated using components reported in Penman (1948).

Productivity

Potential_Production

Century and G-Range use a staged approach to calculate production, first calculating potential production that includes some modifiers (e.g., light, shading), then incorporating further modifiers (e.g., nitrogen availability) to yield estimated production. This subroutine is involved in the first portion of those steps. Biomass for the surface litter, aboveground live biomass, belowground live biomass, and standing dead biomass are computed for the facet. Total biomass is then calculated, and woody biomass as well, with that trimmed to 5000 g/m² if necessary. Temperatures that incorporate shading are calculated, and soil surface temperature is estimated based on day length and shading. The routine then calls **Potential** to continue calculations.

Potential

This subroutine calculates potential production given information calculated in **Potential_Production** and other data. Water available is calculated first, as well as the capacity of the soil to store water. Regression analyses are made to calculate a measure of water availability to plants. A modifier applied to production based on shading is calculated, with that either being zero for main facets (herbs, shrubs, trees), or for plants in the understory:

$$\text{Correction} = 5.0 * \exp (- 0.0035 * \text{Facet_carbon_in_leaves} * 2.5) / \text{Woody_cover}$$

The shading modifier is then $(1 - \text{Woody_cover}) + (\text{Woody_cover} * (\text{Correction} / (\text{Correction} + 1))$

Plant production is then estimated. The restriction based on soil surface temperature and a series of temperature-related parameters passed to the system by the landscape unit parameter file are calculated. The effect of standing dead on potential production is calculated, and live biomass versus dead biomass can reduce potential growth rates. Growth is reduced if incoming shortwave radiation is low, reflecting winter months in the north for example. Production above and belowground is then estimated based on the corrections, then **Grazing_Restrictions** is called to make that functional correction.

Grazing_Restrictions

This subroutine modifies the production of plants based on grazing. Like Century, G-Range uses a series of seven different response rates for the vegetation, which may be reported to the simulation in the landscapes unit parameter file. Codes and effects include:

0 – No effect

$$\text{Aboveground_pot_prod} = \text{Aboveground_pot_prod}$$

$$\text{Belowground_pot_prod} = \text{Belowground_pot_prod}$$

1 – Linear impact of grazing on aboveground potential production

$$\text{Aboveground_pot_prod} = 1 - (2.21 * \text{Fraction_removed_by_grazing}) * \text{Aboveground_pot_prod}$$

$$\text{Belowground_pot_prod} = \text{Root_shoot_ratio} * \text{Aboveground_pot_prod}$$

2 – Quadratic impact of grazing on aboveground potential production and root:shoot ratio

$$\text{Aboveground_pot_prod} = 1 - (2.6 * \text{Fraction_removed_by_grazing}) - (5.83 * (\text{Fraction_removed_by_grazing}^2)) * \text{Aboveground_pot_prod}$$

$$\text{Belowground_pot_prod} = \text{Root_shoot_ratio} + 3.05 * \text{Fraction_removed_by_grazing} - 11.78 * \text{Fraction_removed_by_grazing}^2$$

3 – Quadratic impact of grazing on root:shoot ratio

$$\text{Belowground_pot_prod} = \text{Root_shoot_ratio} + 3.05 * \text{Fraction_removed_by_grazing} - 11.78 * \text{Fraction_removed_by_grazing}^2 * \text{Belowground_pot_prod}$$

4 – Linear impact of grazing on root:shoot ratio

$$\text{Belowground_pot_prod} = 1 - (\text{Fraction_removed_by_grazing} * \text{Grazing_multiplier}) * \text{Belowground_pot_prod}$$

5 – Quadratic impact of grazing on aboveground potential production and linear impact on root:shoot ratio

$$\text{Aboveground_pot_prod} = 1 - (2.6 * \text{Fraction_removed_by_grazing}) - (5.83 * (\text{Fraction_removed_by_grazing}^2)) * \text{Aboveground_pot_prod}$$

$$\text{Belowground_pot_prod} = 1 - (\text{Fraction_removed_by_grazing} * \text{Grazing_multiplier}) * \text{Belowground_pot_prod}$$

6 – Linear impact of grazing on aboveground potential production and root:shoot ratio

$$\text{Aboveground_pot_prod} = 1 - (2.21 * \text{Fraction_removed_by_grazing}) * \text{Aboveground_pot_prod}$$

$$\text{Belowground_pot_prod} = 1 - (\text{Fraction_removed_by_grazing} * \text{Grazing_multiplier}) * \text{Belowground_pot_prod}$$

Aboveground coefficients are not allowed to go below 0.02. Belowground coefficients are not allowed to go below 0.01. At the end of the subroutine, aboveground and belowground potential productions plus root shoot ratio are altered in response to the function selected.

Herb_Growth

This subroutine uses the potential production that has been calculated, as restricts it based on nutrients and other attributes to yield actual growth estimates. If sufficient water is present, and temperature is above 3 degrees C, herbaceous growth is simulated. First, the effect of root interception of nutrients is calculated. Available nitrogen is calculated, then a call is made to **Restrict_Production**, which will limit production based on carbon to nitrogen ratios.

Average total potential production of carbon is calculated, and then the amount of nitrogen fixed is calculated. Growth of leaves and shoots is calculated, after correcting for maintenance respiration. Root growth is simulated, as a fraction of total potential production that remains after shoot growth. Average potential production is calculated, and then nutrients are taken up from a storage pool, if available. Otherwise nutrients are drawn from the soil. Lastly, fixed nitrogen is accounted for. Lignin updates conclude the routine.

Woody_Growth

This subroutine uses potential production that has been calculated, and limits it to actual woody growth as constrained by nutrients. Initially, nitrogen availability is calculated. A value for site potential is calculated, a correction factor adopted in Century and used here, based on precipitation on a landscape cell. Available nitrogen is reduced based on minerals available to savanna trees:

$$\text{Temporary} = \min (\text{Available_nitrogen}, 1.5)$$

$$\text{Correction} = \text{Exp}(-1.664 * \text{Exp} (-0.00102 * \text{Temporary} * \text{Site_potential}) * \text{Tree_basal_area} * \\ \text{Tree_basal_area_to_grass_nitrogen}$$

with the last entry a parameter passed using the landscape units file. Then:

$$\text{Available_nitrogen} = \text{Available_nitrogen} * \text{Correction}$$

If there is sufficient water and temperatures are above zero, production values are estimates based on the impact of root biomass on nitrogen availability. The subroutine then calls **Restrict_Production** to limit production based on carbon to nitrogen ratios. If growth occurs, then the subroutine calculates how carbon should be allocated across woody plant

parts. Changes in carbon and nitrogen are then calculated for the different plant parts. Nitrogen may be taken from storage or the soil, depending upon availability.

Restrict_Production

This subroutine restricts the production of plants based on carbon to nitrogen ratios. First, the available nitrogen is calculated for the cell. Carbon is then converted to biomass (* 2.5 for herbs and leaves, * 2.0 for woody parts). The nitrogen to carbon ratio for plants is then calculated, then repeated just for roots and shoots. The routine then calls **Nutrient_Limitation**, to calculate restrictions on production. Lastly, the routine compares the estimated total production limited by nitrogen, to see if it should be trimmed because of a shortage in stored nitrogen. If the storage pool contains all the nitrogen needed, it is taken from there. Otherwise, only a portion is and the rest is taken from the soil.

Nutrient_Limitation

This subroutine computes the manner in which nutrient limitation may restrict potential production. First, total potential production biomass is estimated for the facet in question. Demand for nutrients is calculated based on a maximum nitrogen to carbon ratio and that total potential production. Updated nitrogen to carbon ratios are calculated. Based on those results and carbon allocation estimates calculated elsewhere, nutrient limitation on total production is estimated.

The system checks to see if production is limited by nutrients, then updates nitrogen uptake for the system. The total production potential limited by nitrogen is updated for each vegetation layer, for the facet of current concern. Lastly, nitrogen taken up by the plants is updated.

Grazing

This routine removes forage that is grazed from a cell. The carbon and nitrogen removed is based on their presence and the fraction of forage removed as described in the landscape units parameter file for the facet in question. Changes to leaf carbon and leaf nitrogen are tallied. Standing dead removed by grazers is then simulated, again based on a passed parameter. Those pools too are tallied. An accumulator for both carbon and nitrogen store the total amount of those materials removed.

The fraction of carbon returned is set based on a parameter in the landscape units file. Nitrogen returned, in contrast, is estimated, following Century. Nitrogen fraction returned is influenced by the clay content in the top layer of the soil:

If clay < 0.0, returned = 0.7

else if clay > 0.3 returned = 0.85

else returned = (0.85 – 0.7) / (0.3 – 0.0) * (Clay – 0.3) + 0.85

The nitrogen in urine and feces is returned in proportion to the rate calculated. Accumulators are tracked, then the material returned to the land is partitioned in litter by calling **Partition_Litter**.

Leaf_Allocation (function)

This function calculated the optimum leaf area index for a stand of trees based on a maximum value and available production. Optimum leaf area index is calculated based on the coarse branch carbon, the maximum value for leaf area index, and a coefficient (k_leaf_area_index) modifier. The value is set to be at least 0.1. From that and a biomass to leaf area index parameter passed to the subroutine from the landscape unit parameters, values are calculated that contribute to a final leaf allocation result. That allocation is trimmed so that it cannot exceed 1.0 or go below 0.01.

Soil and Water

Water_Loss

This subroutine simulates water movements and losses in G-Range, based mostly on H2OLoss in Century. For each rangeland cell, it calculates the ratio of water available to plants versus potential evapotranspiration. If temperatures are below the snow melting point, that is based on melted liquid. More typically, it is calculated based on water available for plant growth and precipitation, divided by potential evapotranspiration.

Water dynamics are then simulated. A call to **Snow_Dynamics** is made. Runoff is calculated using a precipitation threshold and fraction runoff provided in the landscape unit parameter set.

Several lines in the subroutine follow that calculate average live and dead standing biomass, plus the total live biomass and average litter biomass. These values are used to estimate interception of precipitation by vegetation canopies and litter. In this routine, above ground biomass in its use in calculations that follow is trimmed to 800 g/m², and litter is trimmed to not exceed 400 g/m². Following that, canopy interception is calculated:

Interception = (0.0003 * Litter_biomass + 0.0006 * Standing_biomass) *

Scaling_Factor_1

$$\text{Bare_soil_Evaporation} = 0.5 * \text{Exp}((-0.002 * \text{Litter_biomass}) - \\ (0.004 * \text{Standing_biomass}) * \text{Scaling_Factor_2}$$

Total surface evaporation losses are then calculated, and trimmed so as not to exceed 40% of potential evapotranspiration. Water accumulators are then updated.

Potential evapotranspiration water loss is calculated, in cm water per month, unless temperature is below 2 degrees C, in which potential evapotranspiration is 0:

$$\text{Potential_evap_water_loss} = \text{PET_remaining} * 0.65 * (1 - \\ - \text{exp}(-0.020 * \text{Standing_live_biomass}) * \text{CO}_2\text{effects}$$

The simulation then calculates water that is transpired from the water newly added to the cell, and does accounting of changes. Water movement between soil layers is then simulated, with the water added taking soils to their field capacity (if sufficient), then remaining water flowing to the next layer. If the bottom layer is at capacity, the remainder flows into storm flow.

Transpiration water loss across soil layers is then calculated. The wilting point of the soil and the depth of the layer are used to calculate available water for transpiration, weighted by water availability. A measure used in that processes is calculation of relative water content:

$$\text{Rel_Water_Content} = (\text{Water_available} / (\text{Soil_layer_depth} * \\ \text{Wilting_point})) / (\text{Field_Capacity} - \text{Wilting_Point})$$

Water available for plant growth is then calculated, adding the new water input into the cell and not transpired.

Lastly, water content that is evaporated from the top layer of the soil is calculated. The proportion of that water is at least 25% of relative water content. Evaporative loss is calculated as:

$$\text{Evaporation} = 0.25 * \text{PET_top_soil} * \text{Avg_bare_soil_evap} * 0.10$$

Relative water content is recalculated.

Nitrogen_Losses

Nitrogen leaching is related to the movement of water between soil layers and the presence of mineral nitrogen. A parameter controlling minimum leaching is set to 18 in G-Range, the default value in Century. From that, a temporary leaching value is calculated:

$$\text{Leaching_prop} = \min(1.0 - (\text{Min_leach} - \text{Water_Movement}) / \text{Min_leach}, 1.0)$$

Then that value is corrected for texture:

$$\text{Leaching} = 0.2 + 0.7 * \text{Sand} * \text{Mineral_nitrogen} * \text{Leaching_prop}$$

If the layer is the last soil layer, nitrogen lost to storm flow is calculated. Lastly, volatilization is computed, based on a rate of nitrogen volatilization provided in land unit parameters.

Snow_Dynamics

Snowpack is stored in water equivalents. This subroutine adds precipitation to snow if the temperature is below zero.

Rain on snow is incorporated. Then water from the snowpack is sublimated:

$$\text{Sublimated} = \text{Potential_evapotranspiration_remaining} * 0.87$$

Then the amount sublimated is not allowed to exceed the snowpack or go below zero. Sublimation is then removed from snow, and evaporation is increased to reflect snow sublimated.

Simulation of melting of snow is then done. If there is snow, and temperature is above a melting temperature, the quantity of snow melted is estimated using a regression:

$$\text{Melt} = \text{Melting_slope} * (\text{Temperature} - \text{Melting_temperature}) *$$

$$\text{Shortwave_radiation}$$

where shortwave radiation is calculated in a separate function. Melted snow is added to the soil, and snow and water accumulators are tracked.

Plant Part Death

Plant_Part_Death

Simulating plant part deaths is fairly straightforward, and reliant upon a series of parameters provided to the simulation in the landscape unit parameter file that describe how rates of death relate to various stresses. First, herbaceous roots are killed, if temperature is above 0. A relative water index is multiplied by the maximum root death to yield a death rate, which is trimmed to be between 0% and 95%. The quantity of carbon and nitrogen in the dead material is calculated, based on the quantity of those elements and the death rate. Flows to plant parts are simulated. Quantities of

dead plant parts are tallied in G-Range, and used in decomposition, which is slightly different than in Century, which has a scheduled flow design. After tallying changes in carbon and nitrogen, **Partition_Litter** is called, dividing carbon and nitrogen in the new addition to litter based on its fraction of lignin. Respiration flows are then tallied.

Death of leaves and shoot are then simulated. The rate of death is different for herbs that are in the 4th stage of phenology (see SAVANNA) and are part of the proportion that are annuals. Otherwise the main effect on death is a function of water availability and a nominal death rate. If the month is when annual plants are certain to die, death rate for that portion of the plants is increased by that amount. After that, the simulation follows similarly to the death of roots. An exception is that litter is not partitioned. In G-Range, unlike in Century, dead standing carbon is accumulated, and later in the routine, that material is partitioned to litter. So in general, leaves die but do not drop in at this point in G-Range, adding to standing dead. Then standing dead joins litter at a rate the user may control.

The death of seeds are simulated in a way that is roughly analogous to the death of roots. The seeds that die are those that fail to germinate. No adjustment is made to respiration, given that released seeds do not play a role in that process.

This subroutine then calls the **Woody_Plant_Part_Death** subroutine. That simulates the death of leaves, fine branches, coarse branches, fine roots, and coarse roots of shrubs and trees.

Lastly, material in dead standing carbon is adjusted based on a rate of standing dead fall given in the land unit parameters, and the materials are partitioned to litter.

Woody_Plant_Part_Death

Leaf death rate is calculated first, again dependent upon several parameters passed to the system using the landscape units parameter file. As in Century, this subroutine begins with a set of calculations that identify if deciduous leaves should die at a higher rate than the nominal rate. Those calculations depend upon average temperature, whether day length is increasing, and if the cell is generally in a winter period. If it is winter, death rates are increased to account for the proportion of trees that are deciduous. Also, if water is scarce, the death rate is increased based on the proportion of plants identified as drought deciduous.

Following setting the death rate of leaves, adjusted to be per month, carbon death is tallied, and nitrogen death rate is reduced by a fraction of nitrogen that is translocated in woody plants. This is added to stored nitrogen. Dead nitrogen is then tallied. Respiration flows are adjusted based on the proportion of carbon that died, but partitioning to litter does not occur at this point, as the dead leaves and shoots are added to standing dead carbon. Respiration flows are updated.

The death of seeds from woody plants is simulated in a manner analogous to the other plant types. Quantities of dead carbon and nitrogen are calculated, then changes to pool are made, and the dead carbon and nitrogen are partitioned to litter. Seeds do not play a role in respiration. Again, fine branches are simulated in a manner analogous to other plant types. The death of carbon and nitrogen are quantified, and then tallied in the different pools. Fine branches add to dead standing carbon, then later to the litter, like leaves and shoots. Fine roots, coarse roots, and coarse branches follow the typical pattern, in these cases contributing to litter directly.

Whole Plant Dynamics

Whole_Plant_Death

Whole plant death is simulated in a fairly straightforward way. Death is not included when plants are dormant due to cold. Otherwise, for each facet, a series of calculations are made that modify a nominal death rate. That death rate is set by converting the monthly value provided in the landscape unit parameter file to a monthly value. The value is then modified by using a linear regression relationship between the ratio of water to potential evapotranspiration and coefficients that describe death rate effects associated with that. For example, a parameter may state that Ratio_water_PET (none of these are the names of actual parameters, they are used for brevity and clarity) is related to death rate increases as 0.0, 0.10, 20.0, 0.0. These four values are used in a regression. If Ratio_water_PET is 20.0, there is ample water and mortality does not increase in response to water limitations. If Ratio_water_PET is near 0, death rates increase by 10%. Intermediate values yield intermediate increases in death rates:

$$\text{Death_rate} = \text{Nominal rate}$$

$$\text{Death_rate} = \text{Death_rate} + \text{Linear_regression} (\text{Ratio_water_PET}, 0.0, 0.10, 20.0, 0.0)$$

This same logic continues for the effect of grazing on death rates, and the effect of shading on death rate. Death of annuals is included if the month is appropriate, and the facet is herbs.

Facet cover is recalculated following the death of plants. The total population of plants that remain is multiplied by the area of each plant, and then divided by the reference area, 1 km².

Plant_Reproduction

This subroutine uses a set of linear relationships in a fashion similar to **Whole_Plant_Death**. The procedure begins by setting a relative establishment rate for each of the six types of vegetation (3 facets, with 3 in the understory, herbs under shrubs, herbs under trees, and shrubs under trees). This is set based on relative seed production in the landscape unit parameter file. Relative establishment is adjusted using a linear relationship with the ratio of water availability to potential evapotranspiration. Litter carbon reduces relative establishment further, reflecting interference from litter. A similar correction is made based on a linear relationship to herbaceous fine root carbon. Relative establishment was reduced a final time by woody cover and its shading influences.

Potential plants to be established in then calculated by multiplying the population of each plant type with its relative establishment. That allows the current population to influence the rate of establishment, so that common plants are more likely to become established than a rare type. Total populations are then increased by the number of plants established. Trees take precedence in this process, and are not allowed to expand beyond their maximum potential population. Shrubs are then calculated similarly, and are not allowed to expand beyond the available area, with the area occupied by trees excluded. The logic repeats for herbs. Bare ground cover is updated by subtracting the main facet areas from 1.0.

Decomposition

Decomposition

This subroutine begins by partitioning litter among the three facets, based on the facet cover and parameters provided in the landscape unit parameter file. The routines are stable in Century and SAVANNA; readers are referred to those sources for details. Dead fine root carbon and nitrogen are calculated based on the quantity of litter coming into the facet. After calculating the fraction of the material that is lignin, the dead carbon and nitrogen are passed to Partition_Litter, along with the fraction lignin. Based on lignin concentration, that routine allocates the dead material among carbon and nitrogen pools (e.g., slow, fast). Seed decomposition is calculated similarly. Dead fine branches, coarse branches, and roots are then tracked. These carbon and nitrogen totals are stored in accumulators, rather than being immediately partitioned to litter.

The subroutine calls **Effects_on_Decomposition**, which calculates how temperature, anaerobic conditions, and water availability influence decomposition. For each layer, surface and soil, carbon and nitrogen are allocated to fast, intermediate, and passive pools. Changes in mineral nitrogen are tallied. Metabolic decomposition is simulated, based on a rate provided in the landscape unit parameters. Changes to fast metabolic pools are simulated, followed by the intermediate pool, and the passive pool. As in other parts of G-Range and Century, changes in carbon and nitrogen are calculated, then the changes in the accumulators are made. The fraction lignin is calculated, then a call to **Track_Lignin** is used to partition components. In the intermediate pool, decomposition contributes to the fast and passive pools. Invertebrate decomposition is simulated, and then the decomposition of woody plant parts is simulated. Nitrogen fixation is the final part of the subroutine.

Partition_Litter

This subroutine partitions litter into structural and metabolic components, based on the lignin to nitrogen ratio. Nitrogen content is calculated, and lignin concentration is used to calculate a fraction of materials to put to the metabolic pool. That fraction is at least 0.2, leaving 0.8 for the structural pool. Litter structural and metabolic carbon and nitrogen are updated. Plant lignin fraction is updated based on the change simulated within the procedure.

Effects_on_Decomposition

This subroutine is a straightforward calculator of effects on decomposition. It uses linear regression relationships with a series of things that influence decomposition. Temperature effects are calculated first, with decomposition effect standardized to 1 at 30 degrees C. Relative water content effects are calculated. Anaerobic effects on decomposition are calculated. Lastly, the three limits are multiplied to yield a combined effect on decomposition rates.

Track_Lignin

The subroutine tracks lignin concentrations in materials, which in turn influence the fraction of materials put to fast versus intermediate decomposition pools.

Miscellaneous

Each_Year

This subroutine zeros a suite of variables that are not intended to continually accumulate across a simulation. Examples zeroed-out include annual evapotranspiration, shrub and tree carbon accumulators, dead plant parts, excluding standing dead, evaporation, annual respiration, and fixed nitrogen. Standing dead is a stock maintained across years, with dead material flowing into litter at an assigned rate.

Prior to that, the routine recalculates carbon allocation among woody plant parts given the total carbon available. The routine then calculates the proportion of residue that is lignin for each facet, at the surface and in the soil.

Update_Vegetation

For the cell in passed to the subroutine, basal area of trees is updated. A water function is then updated:

$$\text{Water_function} = 1 / (1 + 4 * \exp(-6 * \text{Relative_water_content})).$$

Total accumulators are summed for attributes that are shared between pools. These include:

$$\text{Surface_total_litter_carbon} = \text{Surface_litter_structural_carbon} + \text{Surface_litter_metabolic_carbon}$$

$$\text{Soil_total_litter_carbon} = \text{Soil_litter_structural_carbon} + \text{Soil_litter_metabolic_carbon}$$

$$\text{Surface_total_litter_nitrogen} = \text{Surface_litter_structural_nitrogen} + \text{Surface_litter_metabolic_nitrogen}$$

$$\text{Soil_total_litter_nitrogen} = \text{Soil_litter_structural_nitrogen} + \text{Soil_litter_metabolic_nitrogen}$$

The phenology of plants are updated, based on heat accumulation and a regression relationship that uses parameters included in the landscape units parameter file. Leaf area index is estimated based on leaf carbon for each facet. Lastly, total soil carbon and the ratio of carbon to nitrogen are updated.

Wrap_Up

This subroutine is not involved in simulating ecosystem processes. It writes a file, using many calls to One_Out, that reports the number of times a cell goes below zero or exceeds a very large value. The routine also writes-out a file reporting the time required to complete a simulation.

Progress

This subroutine is not involved in simulating ecosystem processes. It prints the year, month, and progress of a simulation to the screen.

One_Out

This subroutine is not involved in simulating ecosystem processes. It writes to a file a single line that includes the number of times a cell exceeds a given value.

Each_Month

This lengthy subroutine has a simple structure, with dozens of tests to see if a value for a cell has gone below zero and shouldn't or has exceeded a very large number. If those conditions occur, the value is reset to zero or the large number and a structure that tallies those errors is incremented. Near the end of the subroutine, the fraction of live and dead grazing that is removed per cell is updated.

Linear (function)

Based on a routine within SAVANNA, Linear looks up a y value provided an x value, for any number of pairs of points defining a (perhaps broken) line.

Line (function)

This subroutine implements a simple look-up of a y value based on an x value and points along a line.

Output**Initialize_Outputs**

The subroutine is not involved in simulating ecosystem processes. It opens a copy of each output file, ensuring it was empty rather than being appended to, and writes header information.

Output_Surfaces

The subroutine is not involved in simulating ecosystem processes. The subroutine is a series of calls to the Output_One_Surface, producing a month of output in each of the output types produced in G-Range.

Output_One_Surface

The subroutine is not involved in simulating ecosystem processes. The subroutine writes a spatial surface for a given month to the output file passes to the subroutine.

Appendix G. Hilinski used Doxygen, an automatic documentation tool, to create a set of documentation pages for a version of G-Range from August 10, 2011. The main page is shown below. A great deal of detail is shown in the pages to which it connects. In the beta distribution of G-Range, the pages are stored under \GRange\Auto. Minor changes have been made to G-Range since its analysis by Doxygen, mostly to fire and fertilization modeling.

