Version: 8-5-2018

Development of ECOTRAN models and their analysis:

This is a very brief summary of the ECOTRAN end-to-end modeling code. See also the comments within the code files themselves. It is being provided for others to use, adapt to their own model studies, and to expand.

The time-dynamic simulation coding (see section "6: Time-dynamic simulations") was used in our study "Understanding large-scale energy flows through end-to-end shelf ecosystems - the importance of physical context" by Ruzicka, Steele, Brink, Gifford, and Bahr (Journal of Marine Systems, in press August 2018).

Please also see in the references section these other publications that have used ECOTRAN: Steele and Ruzicka 2011; Ruzicka et al. 2012; Ruzicka et al. 2013a,b,c; Robinson et al. 2014; Robinson et al. 2015; Ruzicka et al. 2016a,b; de Haast et al. 2017; Chiaverano et al. 2018.

NOTE: Please be aware that this code suite is frequently being revised for organizational clarity, robustness (esp. to alternate food web functional group types), functionality, and error correction. There may be errors that I have not caught or that appear in model scenario situations that I have never tried.

Please address questions to Jim Ruzicka (jim.ruzicka@oregonstate.edu), and I will do my best to provide clarifications.

Organization of the summary:

1: Definition of model parameters

ECOPATH-style mass-balance models are set up within excel. For the test model, TESTmodel_07252018.xlsm, VisualBasic code provides the ECOPATH (ecopath.org) algorithms to balance the model directly. This is made available to us from Kerim Aydin (NOAA AFSC).

After all parameters are set up, there is a big red button to run ECOPATH on the *MAIN* tab (scroll to the right to see the button). The second big red button is to write the entire model to a .csv file for use by ECOTRAN in matlab (TESTmodel_07252018.csv).

The first five tabs are where the standard ECOPATH parameters are defined: *MAIN*, *Diets*, *Detritus*, *Fishing*, *Discards*.

The next five tabs are where various parameter uncertainty terms are defined for use in generating Monte Carlo models (if wanted, its optional).

The next two tabs are where the results from ECOPATH are returned: *MainOutputs*, *Mortalities*.

The final three tabs are for the definition of ECOTRAN parameters. 1) *EcotranType* tab is where the type of each functional group is defined by number code, and where aggregated groups can be defined for automated aggregation (by function f_AggregateResults_EwE_03122015). 2) *EcotranRecycling* tab is where the fates of detritus and NH4 excretion are defined. There are two detritus types: feces & non-predation mortality. Each detritus type goes to one of two terminal pools: surface or benthic. Note that there may be many detritus pools defined as functional groups in ECOPATH, but ultimately, all detritus that is not consumed will wind up in either the terminal pelagic or the terminal benthic detritus pool. Similarly there are two NH_4 pools: pelagic NH_4 and benthic NH_4. This tab is also where nitrate and ammonium production is partitioned between primary producers (on the left) 3) The *FunctionalResponse* tab is where functional response parameters are defined. There is room for expansion in this tab to include many optional functional response parameters; as of now, code only uses the first column.

OPTIONAL: After a balanced model has been generated, there is code available to generate and evaluate Monte Models (see section “3: Generate Monte Carlo models”)

2: Assembly of the ECOTRAN EnergyBudget_matrix and BioenergeticBudget

The balanced ECOPATH model built in the excel VisualBasic .xlsm file is exported directly to a .csv file. We now switch to Matlab (www.mathworks.com).

To run ECOTRAN static scenarios, dynamic models, making web plots, or calculating various model metrics, a common set of steps is followed (see code ECOTRAN_StaticScenarios_TEST_08052018 where all these steps are combined):

Step 1 - Load EwE results from .csv file into memory & aggregate functional groups

[dat]	= f_read_EwE_csv_04292016(readFile)
[EwEResult, PEDIGREE]	= f_AggregateResults_EwE_03122015(dat)

Functional groups do not need to be aggregated beyond the level at which the ECOPATH model was defined, but the function f_AggregateResults_EwE_03122015 is still needed in order to prepare the ECOPATH parameters for later construction of the ECOTRAN model.

Step 2 (optional) **-** Load pre-generated Monte Carlo models. Monte Carlo models are generated in other code. Monte Carlo models can be generated from directly the ECOTRAN EnergyBudget_matrix (Method 1; this is a new method and preferred because it is fast). Or, pre-generated Monte Carlo models can be loaded now from a .mat file. (Method 2; no longer supported). (See section "3: Generate Monte Carlo models")

Step 3 - prepare the end-to-end ECOTRAN model

[ECOTRAN] = ECOTRANuncertainty_05062016(EwEResult, MonteCarloStore)

calls sub-functions:

ECOTRANuncertainty_05062016	Generate an ECOTRAN (E2E) model
f_ECOfunction_05142015	returns a single ECOTRAN model for 1 "type" EwE model or 1 MonteCarlo EwE model
f_RedistributeCannibalism	remove cannibalism terms on diagonal of matrix EwE_diet
f_CalcPredationMatrix	matrix defining the fraction of total predation on each producer, p, going to each consumer, c

The ECOTRAN structure variable has a lot of information in it, but the main outputs are the EnergyBudget_matrix and the BioenergeticBudget.

EnergyBudget_matrix (A_{cp})--- this is the heart of ECOTRAN. This matrix defines the fate of all energy (biomass) that enters a functional group box. Groups as producers run across columns. Groups as consumers run down the rows. Each column sums to 1. Note that if there is net immigration of a group into the model domain (emigration fraction is negative), then the entries in all other rows of that column will be greater to include distribution of immigration fate.

- Any group may be referred to in this text as either producer (p) or a consumer (c) depending upon your frame of reference within the food web.
- Primary producers are treated as consumers of nutrients (Consumer 1 in the example matrix below).
- Fisheries are treated as any other predator – except that the column sum of fisheries do not sum to 1. Fisheries remove production from ecosystem and only return a portion of that take as discard contribution to detritus.
- Bacteria can be included explicitly as a defined functional group, or bacteria can be defined implicitly (as in the example below) as flow from detritus to NH$_4$ pools.
- Unconsumed pelagic detritus flows to benthic detritus. Column sum of benthic detritus does not need to sum to 1 – as this and fisheries are the ultimate loss pathways out of ecosystem. (alternatively, benthic detritus column sums to 1 but transfer efficiency is < 1)
- Advection losses and gains are not accounted for within the EnergyBudget_matrix. These gains & losses are taken or added directly to production rate estimates

	NO3	pelagic NH4	benthic NH4	Consumer 1	Consumer 2	Fishery 1	Fishery 2	pelagic detritus	benthic detritus
NO3									
pelagic NH4					0.05			0.08	
benthic NH4					0.2				0.25
Consumer 1	1	1	1						
Consumer 2				0.42					
Fishery 1				0.3	0.05				
Fishery 2				0.05	0.3				
pelagic detritus				0.2					
benthic detritus				0.05	0.2	0.1	0.03	0.82	
emigration					0.1				

Table 1. Format of the ECOTRAN EnergyBudget_matrix.

These are the array variables that describe the static, balanced state of the average biomass flows over time and bioenergetics of individual functional groups:

BioenergeticBudget --- summary of the fate of all energy (biomass) that enters a functional group box. Basically, a simplified, 3-row version of the EnergyBudget_matrix. The sum of each column = 1, except for fisheries & benthic detritus):

row 1 = BioenergeticBudget_production
row 2 = BioenergeticBudget_feces
row 3 = BioenergeticBudget_metabolism

BioenergeticBudget_ProductionDetail --- a more detailed breakdown of row 1 (BioenergeticBudget_production) in the BioenergeticBudget. The sum of each column = BioenergeticBudget_production.
row 1 = BioenergeticBudget_SumPredation (total consumption of each group p going to all its consumers)
row 2 = BioenergeticBudget_eggs (total consumption of each group p going to eggs, gametes, or live births)
row 3 = BioenergeticBudget_OtherMortality (total consumption of each group p going to senescence "other mortality")
row 4 = BioenergeticBudget_BA (total consumption of each group p going to biomass accumulation)
row 5 = BioenergeticBudget_EM (total consumption of each group p going to emigration)

BioenergeticBudget_OtherMortDetail --- fraction of "other mortality" flowing to either pelagic or to benthic detritus pools. Each column sums to 1.
row 1 = flow to terminal pelagic detritus
row 2 = flow to terminal benthic detritus

BioenergeticBudget_FecesDetail --- fraction of feces production flowing to either pelagic or to benthic detritus pools. Each column sums to 1.
row 1 = flow to terminal pelagic detritus
row 2 = flow to terminal benthic detritus

3: Generate Monte Carlo models

There are two methods to generate alternate food web models from pre-defined estimates of uncertainty about all physiological and diet preference parameters within ECOTRAN. There is also a function for generating alternative predator-prey functional response parameters (see section 6: Time-dynamic simulations, function f_FunctionalResponse_MonteCarlo_09122016)

Method 1: (NOTE: The preferred method)

function: f_E2E_pedigree_08032018 calculate the uncertainty for every cell within the EnergyBudget_matrix (A_{cp}) from pre-defined uncertainty values for all parameters.

function: f_E2E_MonteCarlo_08032018 generate a stack of random EnergyBudget_matrices. The first model in the stack is the “type” model generated from the VisualBasic mass-balanced food web model.

Generate a set of Monte Carlo models directly from the end-to-end (E2E) ECOTRAN EnergyBudget_matrix (A_{cp}). The first step here is to define uncertainty values (Coefficients of Variation) for all model parameters and from that information calculate the uncertainty for every cell within the EnergyBudget_matrix (A_{cp}). This is done in function f_E2E_pedigree_08032018. The second step is to generate a stack of random EnergyBudget_matrices. This is done in code f_E2E_MonteCarlo_08032018. Parameter uncertainties (“pedigrees”) may either be defined individually within the VisualBasic.xlsm file (e.g., TESTmodel_07252018.xlsm), or they may defined as default values more generally by parameter type in the scenario code (as is done here in example code ECOTRAN_StaticScenarios_TEST_08052018).

To briefly summarize what happens here. We first establish uncertainty values about each of the values of the BioenergeticBudget for each model group. These terms define the fate of total consumption: feces, metabolism, egg production, predation, senescence, biomass accumulation, and emigration into/out of model domain. The uncertainties about some of these terms may be well-defined (e.g., assimilation efficiency for feces production) and some are very poorly-defined (e.g., predation and senescence). We then define the uncertainty parameters for all members of the EnergyBudget_matrix. The EnergyBudget_matrix is essentially a more detailed version of the BioenergeticBudget. The EnergyBudget_matrix further breaks down the predation term for each producer group into individual predators that eat that producer (or primary producers as consumers of a nutrient pool). It also breaks down feces and senescence fates into various detritus pools. Once uncertainties (CV) are defined for the BioenergeticBudget and the EnergyBudget_matrix, a random version is generated for both of these matrices. The elements of the random BioenergeticBudget are adjusted so that column totals for every model group is equal to ONE. This adjustment is done hierarchically so that well-defined terms (e.g., feces, eggs, metabolism) are minimally adjusted while poorly-defined terms (e.g., predation and senescence) can be more broadly adjusted. Some individual terms may be greater than 1 or even negative (e.g., biomass accumulation, emigration, predation), while physiological terms terms must be between 0 and 1, but the total of each column must sum to ONE. A random EnergyBudget_matrix is then generated and the individual elements within the EnergyBudget_matrix are adjusted to match the random BioenergeticBudget. For example, all consumer elements in any given column of the EnergyBudget_matrix are scaled so that their sum is equal to the predation sum of the BioenergeticBudget. The sum of each column of the EnergyBudget_matrix is then also equal to ONE.

The theory behind this method and why it is fast. As each element of the EnergyBudget_matrix is mathematically determined by various defined physiological, diet, predation, senescence rate, population growth, and emigration terms and we have an uncertainty value (CV) defined for each of these terms, then we can calculate the uncertainty level for each element within the EnergyBudget_matrix (see notes in code file for rules of adding and multiplying uncertainty terms and references). Because the sum of each column within the EnergyBudget_matrix sums to one, the model is in thermodynamic balance (see Steele 2009). This makes the method 1 technique fast. We do not generate a random ECOPATH-style (EwE) model from individual parameters and skip having to evaluate whether it is in mass-balance (all ecotrophic efficiency terms <= 1).

NOTES:

- The method 1 code works for generating **normally** distributed elements. The steps for generating **uniformly** distributed elements is also in the function but have not been fully checked through.

Method 2: (NOTE: very slow, no longer supported)

Models are randomly generated by drawing from the uncertainty distributions of individual physiological and diet parameters. Each model is evaluated for mass-balance and only balanced models are retained. Because potential un-balanced models far outnumber potential balanced models, this is a very time-consuming technique. An example set of Monte Carlo models is provided for the test model TESTmodel_07252018.xlsm (file TEST_MCnorm_18-Jan-2015.mat).

function: EwE_MonteCarlo_01182015 main code

calls sub-functions:

f_read_EwE_csv_02022016	read in ECOPATH (EwE) model as variable 'dat'
f_AggregateResults_EwE_03122015	prepare EwE model for use by ECOTRAN; also, aggregate functional groups here if wanted
f_EwEinterval_MonteCarlo_04092014	prepare sampling interval
f_EwEnormal_MonteCarlo_04092014	generate 1 random model
f_DietPreference_Readjust_04052014	adjust diets w/ prey guilds
f_EE_MonteCarlo_04092014	evaluate model balance based on pre-defined EE limits of each group, NOTE: allows for “out of balance models” if wanted

Step 1 - Prepare EwE file as a .csv file as in "1) definition of model parameters"

Step 2 - In matlab code EwE_MonteCarlo_01182015:
Define SaveFile_directory and saveFile
Define ReadFile_directory and readFile

Step 3 - Pre-test difficulty of generating mass-balanced random model by adjusting the variables PEDIGREE.SamplingScale, PEDIGREE.SamplingScale_Diet, and PEDIGREE.SamplingScale_Fisheries. Random models are generated by sampling from within a normal distribution defined by your CV parameter pedigrees. By default, sampling is restricted to within 1 STD of the original parameter value (you can extend or shrink this by changing PEDIGREE.STD_scaler. The problem is that for some models, millions and millions of models are tossed before 1 good model is found. So, as a compromise in order to be able to get a reasonable set of Monte Carlo models from a laptop computer, you can "steepen" the normal distribution about which random parameter values are drawn. The sampling range doesn't change (still 1 STD be default), but the random sampling becomes more concentrated near the original parameter value as you increase the value of PEDIGREE.SamplingScale, PEDIGREE.SamplingScale_Diet, and PEDIGREE.SamplingScale_Fisheries. I aim to get at least 1 good model for every 50,000 random models. The justification for this compromise is that the number of models generated is so large you are still sampling over the entire parameter range even if you a focusing near the original parameter set values.

NOTES:

- If you are aggregating a model down to a smaller number of boxes, I find it best to make sure that your aggregated model is in balance before you run the Monte Carlo. Computer-aggregated models may not necessarily be in automatic balance.
- This code corrects for the potential of drawing physiology parameters that allow an implied "negative" metabolic rate by rejecting those parameter sets.

4: Static model scenarios

Run a static model scenario by altering a portion of the food web. A static model scenario shows the impact of changes in food web structure or changes to the strength of any pathway(s) within the food web. At present, it assumes linear predator-prey relationships. It is essentially the state of time-dynamic model after an indefinite period of time when a new equilibrium is obtained (Collie et al. 2009, Steele 2009).

Static scenarios are built and plotted in code file ECOTRAN_StaticScenarios_TEST_08052018. Here you can…

- Change biomass and consumption rate of one or more consumer groups
- Change predation pressure on one or more specific producer groups
- Change and compare ecosystem driver rates
- Change fishery group landings and/or discard rates

function: ECOTRAN_StaticScenarios_TEST_08052018

calls sub-functions:

f_read_EwE_csv_04292016	read in ECOPATH (EwE) model as variable 'dat'
f_AggregateResults_EwE_03122015	prepare EwE model for use by ECOTRAN; also, aggregate functional groups here if wanted
ECOTRANuncertainty_05062016	Generate an ECOTRAN (E2E) model
f_ECOfunction_05142015	returns a single ECOTRAN model for 1 "type" EwE model or 1 MonteCarlo EwE model
f_RedistributeCannibalism	remove cannibalism terms on diagonal of matrix EwE_diet
f_CalcPredationMatrix	matrix defining the fraction of total predation on each producer, p, going to each consumer, c
f_E2E_pedigree_08032018	Method 1 Monte Carlo function: generate the pedigree (CV) of each cell of the EnergyBudget_matrix (A_{cp})
f_E2E_MonteCarlo_08032018	Method 1 Monte Carlo function: generate Monte Carlo models from the EnergyBudget_matrix (A_{cp})
f_WebProductivityWLoss	calculate production rates of all groups under a given driver (e.g., nitrate or primary production); also accounts for defined rates of group production export when running under static scenarios
f_ScenarioGenerator_08302013	modify the EnergyBudget_matrix according to forced scenario conditions

f_WebProductivityWLoss	
f_CompileScenarioResults_08192013	compile scenario results into a matrix (see notes in ECOTRAN_StaticScenarios code for column definitions)
p_PlotScenarioResults_02092018	plot scenario results as changes in group production relative to unmodified base model
myboxplot_4 & myboxutil_2	(from the Jorn_Diedrichsen_Toolbox; found at www.icn.ucl.ac.uk/motorcontrol/toolboxes/matlab_toolboxes.htm; myboxplot_4 and myboxutil_2 are modified versions; other boxplot functions could be used instead)

These are the steps that take place in the code:

Step 1 - Identify and load the ECOPATH (EwE) mass-balanced model. Define the .csv model filename. This is the mass-balanced model constructed using K. Aydin's VisualBasic version of the ECOPATH algorithms and exported to .csv format. Models from other sources may be used, but they need to be arranged into the same column format as produced by TESTmodel_07252018.xlsm. Also note that the file directory path will need to be updated for the local computer. Two functions are called here: f_read_EwE_csv_04292016 and f_AggregateResults_EwE_03122015.

Step 2 (optional) - Load pre-generated Monte Carlo mass-balanced models (see section "3: Generate Monte Carlo models"). This is Monte Carlo method 2. I've created a new Monte Carlo method (Monte Carlo method 1) that generates Monte Carlo models directly from the ECOTRAN EnergyBudget_matrix (A_{cp}). Method 1 is the preferred method as it is much faster (see code step 4).

Step 3 - Prepare the end-to-end ECOTRAN model (E2E). This is the heart of ECOTRAN and calls function: ECOTRANuncertainty_05062016.

Step 4 - Monte Carlo method 1. This is the preferred method to generate Monte Carlo models because it is fast. (see section "3: Generate Monte Carlo models").

Step 5 - Define the transfer efficiencies for each group. Because the EnergyBudget_matrix defines the fate of all consumption (rather than only of production), we define the transfer efficiencies to all be 1 EXCEPT for the terminal benthic detritus group. This is a poorly defined term for any system. By practice, we define terminal benthic detritus transfer efficiency to be 0.1; the model is fairly insensitive to all but extreme values.

Step 6 - Define production loss fractions. Static model scenarios can account for the fraction of nutrient, plankton, and detritus production that is lost from the model domain via physical advection (e.g., Steele and Ruzicka 2011). These terms are defined as fractions of total group production and are defined here (all set to zero by default).

Step 7 - Initialize InputProductionVector. The model is driven by an InputProductionVector. This can defined at any trophic level in theory, but usually is a nutrient input rate or a primary production rate. By default, the InputProductionVector is defined here as the primary production rate P = b * (p/b) for the primary producer groups. Please note that a base and a scenario vector can be defined for cases where you want to compare the effect of two alternate ecosystem drivers.

Step 8 - Calculate productivity for each functional group within the un-altered food web. This calls function f_WebProductivityWLoss.

Step 9 - Generate a model scenario. There are many options here. The example option is a scaling up of the abundance of one group (baleen whales) and their consumption of model resources at the expense of all other consumer groups that eat the same things (with indirect effects along all trophic chains). First, define the consumer group(s) to be modified. Second, define the amount they should be re-scaled by. Third, define "offset" consumers (those consumers that get to eat more or eat less prey because of the changes to the group with modified abundance. Fourth, define target producers. You can define whether the modified group is changing its consumption of just one, a few, or all of its prey groups. The food web (EnergyBudget_matrix) is modified by code f_ScenarioGenerator_08302013. All Monte Carlo versions of the food web are similarly modified. Note that there will be a warning given for an individual Monte Carlo model in the case that one or several prey groups cannot support the changed abundance of the modified consumer. The modified models will still be in thermodynamic balance but the target change to the consumer biomass will not be realized (because there is not enough food available to support that amount of change). Results are expressed as the change in group production relative to the base model: $(P_{scenario} - P_{base}) / P_{base}$. Results are compiled into table form by function f_CompileScenarioResults_08192013. Results are expressed graphically (box plots) by function p_PlotScenarioResults_02092018.

5: Footprint, reach, and web plots

Two useful metrics that can be derived from a static food web model are the footprint and the reach:

FOOTPRINT fraction of each PRODUCER's production required to support a particular CONSUMER group.

REACH fraction of particular PRODUCER group production going to support each CONSUMER group.

function: FootprintReach_TEST_07262018

calls sub-functions:

f_read_EwE_csv_04292016	read in ECOPATH (EwE) model as variable 'dat'
f_AggregateResults_EwE_03122015	prepare EwE model for use by ECOTRAN; also, aggregate functional groups here if wanted
ECOTRANuncertainty_05062016	Generate an ECOTRAN (E2E) model
f_ECOfunction_05142015	returns a single ECOTRAN model for 1 "type" EwE model or 1 MonteCarlo EwE model
f_RedistributeCannibalism	remove cannibalism terms on diagonal of matrix EwE_diet
f_CalcPredationMatrix	matrix defining the fraction of total predation on each producer, p, going to each consumer, c
f_E2E_pedigree_08032018	Method 1 Monte Carlo function: generate the pedigree (CV) of each cell of the EnergyBudget_matrix (A_{cp})
f_E2E_MonteCarlo_08032018	Method 1 Monte Carlo function: generate Monte Carlo models from the EnergyBudget_matrix (A_{cp})
f_Footprint_07272018	calculate footprint metrics for target group consumer. Excludes phytoplankton, micrograzers, and microbes from denominator in system footprint calculation.
f_ProductionTrace_07272018	trace producer group (p) production through static model food web. Code blocks nutrient recycling in ALL cases. Code automatically tests if target group consumer is a detritus group and blocks trophic recycling if target group consumer is anything but detritus.
f_Reach_01212018	calculate reach metrics

f_DietTrace_03152015	trace source group flow through static model food web
f_DietTraceDownward_03152015	trace source group flow through static model food web
p_WebPlotter_01032017	Plot a food web diagram.

These are the steps that take place in the code:

(NOTE: At present (8/2018), Monte Carlo analyses do NOT work for the footprint and reach metrics. Code to do this is currently being error-checked).

Step 1 - Identify and load the ECOPATH (EwE) mass-balanced model. Define the .csv model filename. This is the mass-balanced model constructed using K. Aydin's VisualBasic version of the ECOPATH algorithms and exported to .csv format. Models from other sources may be used, but they need to be arranged into the same column format as produced by TESTmodel_07252018.xlsm. Also note that the file directory path will need to be updated for the local computer. Two functions are called here: f_read_EwE_csv_04292016 and f_AggregateResults_EwE_03122015.

Step 2 (optional) - Load pre-generated Monte Carlo mass-balanced models (see section "3: Generate Monte Carlo models"). This is Monte Carlo method 2. I've created a new Monte Carlo method (Monte Carlo method 1) that generated Monte Carlo models directly from the ECOTRAN EnergyBudget_matrix (A_{cp}). Method 1 is the preferred method as it is much faster (see code step 4).

Step 3 - Prepare the end-to-end ECOTRAN model (E2E). This is the heart of ECOTRAN and calls function: ECOTRANuncertainty_05062016.

Step 4 (optional) - Monte Carlo method 1. This is the preferred method to generate Monte Carlo models because it is fast. (see section "3: Generate Monte Carlo models").

Step 5 - Calculate footprint metrics. First, identify one consumer group that you want to calculate footprint metrics for. Second, decide whether you want to include the costs of non-production consumption (feces and metabolism costs) along each trophic transfer step. The default is "no". Third, call function f_Footprint_07272018 to calculate the fraction of production by each producer group flowing to the target group consumer. The code blocks nutrient recycling in ALL cases. The code automatically tests if the target consumer group is a detritus group and blocks trophic recycling of detritus if the target consumer group is anything but detritus. Three variables are returned:

TraceGroup_FootprintVector: The fraction of each producer group flowing to the target consumer. (2 horizontal vectors: row 1 = mean across MonteCarlo models, row 2 = std across MonteCarlo models, columns = producer groups). This variable is used for web plotting of footprint box colors.

TraceGroup_SystemFootprint: SYSTEM-LEVEL footprint = ratio of total target consumer group footprint on all producers to total production of all consumer groups. (2 horizontal vectors: row 1 = mean across MonteCarlo models, row 2 = std across MonteCarlo models, columns = producer groups). (NOTE: prior to 2018, denominator of ratio was formerly total production of all producer groups)

FootprintArray: Fraction of each producer group flowing to each consumer. (3D stack of matrices; rows = consumers, columns = producers, layers = MonteCarlo models)

Fourth, rearrange group orders so that groups of the footprint and reach metrics are in identical order. This is necessary for food web plotting Step 7, otherwise it can be omitted.

Step 6 - Calculate reach metrics. First, identify one producer group that you want to calculate reach metrics for. Second, call function f_Reach_01212018 to calculate the fraction of production by the target producer group going to each consumer group. Six variables are returned:

TraceGroup_ReachVector: fraction of production by target producer group going to each consumer group. (2 vertical vectors: rows = producer groups, column 1 = mean across MonteCarlo models, column 2 = std across MonteCarlo models)

TraceGroup_SystemReach: SYSTEM-LEVEL reach = ratio of total production by target producer group going to all consumers over total production of all consumer groups. (Horizontal vectors: row 1 = mean across MonteCarlo models, row 2 = std across MonteCarlo models).

DietTrace_upward: Fraction of energy within each linkage ultimately originating from target producer group; mean across all MonteCarlo models. (2D matrix: rows = producers, columns = consumers). This variable is used for web plotting reach pathway colors.

TraceFraction_upward: Fraction of each CONSUMER production originating from target producer group; mean across all MonteCarlo models. (horizontal vector: 1 X clms of consumers). This variable is used for web plotting reach box colors.

DietTrace_downward relative contribution of each linkage to production of target CONSUMER group. (2D matrix: rows = producers, columns = consumers). This variable is used for web plotting footprint pathway colors.

TraceFraction_downward: Fraction of target CONSUMER group originating from each PRODUCER group; mean across all MonteCarlo models. (Horizontal vector: columns = consumers).

Step 7 - Call function p_WebPlotter_01032017 to plot a food web diagram. This function plots the food web diagram font label size proportional to functional group production rate, y-axis position equal to trophic level, green shading proportional to the footprint of the target group, and red color proportional to the reach of the target group. First, the x-position must for each box must be defined manually. This is a tedious process of trial-and-error to make sure that group boxes are placed aesthetically and are easy to read. There is no on-screen positioning ability (yet). Second, define y-positions of fisheries and import prey, and re-define y-positions of detritus groups to prevent box overlap, if necessary. Third, add fisheries and import prey box labels, and edit other box labels as necessary.

NOTES:

- The food web plotter uses the groups and order of the ECOPATH (EwE) model (i.e., it includes import diet, does not include nutrients, fisheries are positioned slightly differently than in the ECOPATH order).
- Box size range, trophic path weight, color intensities, and axis range can be re-adjusted in Step one of the p_WebPlotter_01032017 code.
- Turn off Monte Carlo options when using the web plotter. Make sure to activate step 5d to make sure footprint and reach metrics are in the ECOPATH (EwE) group order.

6: Time-dynamic simulations

This code suite is used for ECOTRAN time-dynamic simulations within four different physical settings. ECOTRANdynamic_context_08032018 and ECOTRANdynamic_context_basin_08032018are the main code files for running the time-dynamic simulations in the comparative shelves study ("Understanding large-scale energy flows through end-to-end shelf ecosystems - the importance of physical context").

function: ECOTRANdynamic_context_08032018	(use with upwelling, downwelling, and bank physics)
ECOTRANdynamic_context_basin_08032018	(use with basin physics)

calls sub-functions:

f_read_EwE_csv_02022016	read in ECOPATH (EwE) model from VisualBasic .csv file and store as variable 'dat'; (for VisualBasic food webs with up to 5 primary producers); (NOTE: use f_read_EwE_csv_04292016 for VisualBasic version that allows for 10 primary producers)
f_AggregateResults_EwE_03122015	prepare EwE model for use by ECOTRAN; also aggregate functional groups here if wanted
ECOTRANuncertainty_05062016	Generate an ECOTRAN (E2E) model ; the heart of ECOTRAN
f_ECOfunction_05142015	returns a single ECOTRAN model for 1 "type" EwE model or 1 Monte Carlo EwE model
f_RedistributeCannibalism	remove cannibalism terms on diagonal of EwE_diet matrix
f_CalcPredationMatrix	matrix defining the fraction of total predation on each producer, p, going to each consumer, c
f_E2E_pedigree_08032018	Method 1 Monte Carlo function: calculate the pedigree (CV) of each cell of the EnergyBudget_matrix (A_{cp})
f_TerminalDetritus_08032018	add an ULTIMATE detritus pool; necessary for management of bacterial metabolism of terminal benthic detritus in time-dynamic simulations
f_E2E_MonteCarlo_08032018	Method 1 Monte Carlo function: generate Monte Carlo models from the EnergyBudget_matrix (A_{cp}); model 1 of the stack is the "type" model defined by the parameters of the VisualBasic .csv file
f_OrdinalDate	ordinal date is day of year with January 1 of any year = 1

- f_ECOTRANphysics_upwelling_08022018 physics for upwelling setting
  - round2 round to a specified number of decimal places
  - f_LightIntensity solar light intensity for given location and time
  - calcur_res.mat dataset with monthly mean salinity, temperature, and NO3+NO2 data for NCC upwelling ecosystem. (see: http://science.whoi.edu/users/seasoar/cse/)
- f_ECOTRANphysics_downwelling_08022018 physics for downwelling setting
  - round2 round to a specified number of decimal places
  - f_LightIntensity solar light intensity for given location and time
  - cgoa_ancyc.mat dataset: monthly mean nutrient observations for CGoA downwelling ecosystem. (see: http://science.whoi.edu/users/seasoar/cse/)
- f_ECOTRANphysics_bank_08022018 physics for bank setting
  - round2 round to a specified number of decimal places
  - f_LightIntensity solar light intensity for given location and time
  - gb_ancyc.mat dataset: monthly mean nutrient observations for GB bank ecosystem. (see: http://science.whoi.edu/users/seasoar/cse/)
- f_ECOTRANphysics_basin_08022018 physics for basin setting; (rates from M. Heath)
  - round2 round to a specified number of decimal places
  - f_LightIntensity solar light intensity for given location and time
- f_FunctionalResponse_MonteCarlo_09122016 prepare functional response terms (vulnerability array); allows generation of random functional response parameters
- f_MichaelisMenten_05152016 (optional) phytoplankton uptake rate of NO_3 & NH_4 (mmole N/m3/d) & p/b @ t as calculated from Michaelis-Menten uptake kinetics
- f_InitialProductionRates_05112016 calculate initial or mean production conditions
  - f_WebProductivityWLoss calculate production rates of all groups under a given driver (e.g., NO_3 or primary production); also accounts for defined rates of group production export when running static scenarios
- f_StaticProductionTimeseries_09042017 calculate initial or mean production conditions for ocean boundaries
  - f_WebProductivityWLoss calculate production rates of all groups under a given driver (e.g., nitrate or primary

production); also accounts for defined rates of group production export when running under static scenarios

f_ECOTRANode_ReflectiveBoundary_05182017 solve the ODE for solution at each time-point; for reflective boundary conditions; (use with upwelling, downwelling, and bank physics)

f_ECOTRANode_ReflectiveBoundary_basin_05242017 solve the ODE for solution at each time-point; for reflective boundary conditions; (use with basin physics)

f_ECOTRANode_DefinedBoundary_08032017 solve the ODE for solution at each time-point; for defined boundary conditions; (use with upwelling, downwelling, and bank physics)

f_ECOTRANode_DefinedBoundary_basin_08032017 solve the ODE for solution at each time-point; for defined boundary conditions; (use with basin physics)

Step 1 - Identify and load the ECOPATH (EwE) mass-balanced model. Define the .csv model filename. This is the mass-balanced model constructed using K. Aydin's VisualBasic version of the ECOPATH algorithms and exported to .csv format. Models from other sources may be used, but they need to be arranged into the same column format as produced by the VisualBasic file.

Choose one food web model at a time. The four food web models are:

NCC_NSFagg_G_8-10-2016_B.csv	Northern California Current
CGoA_NSFagg_E_08102016_B.csv	Coastal Gulf of Alaska
GB_NSFagg_P_02082017.csv	Georges Bank
NorSea_NSFagg_G_9-18-2016.csv	North Sea

Two functions are called in this step: f_read_EwE_csv_02022016 reads the VisualBasic (EwE) mass-balanced food web model from a .csv file and f_AggregateResults_EwE_03122015 prepares this file for work in the ECOTRAN code.

NOTES:

- The file directory path to each food web model will need to be updated for the local computer.
- Trophic aggregation of functional groups can be done by the function f_AggregateResults_EwE_03122015 but it is not necessary. I advise that if automated trophic aggregation is done, the aggregated model be entered into a new VisualBasic file, proofed for mass-balance, and exported as a new .csv file.

Step 2 (optional) - Load pre-generated Monte Carlo mass-balanced models (see section "3: Generate Monte Carlo models"). This is Monte Carlo method 2 and is no longer supported. Monte Carlo method 1 is now the preferred Monte Carlo method. Monte Carlo method 1 generates random models directly from the ECOTRAN EnergyBudget_matrix (A_{cp}). Monte Carlo method 1 is the preferred as it is much faster (see code step 4).

Step 3 - Prepare the end-to-end ECOTRAN model (E2E). This is the heart of ECOTRAN and calls function: ECOTRANuncertainty_05062016. (see "Section 2: Assembly of the ECOTRAN EnergyBudget_matrix and BioenergeticBudget").

Step 4 - Monte Carlo method 1. This is the preferred method to generate Monte Carlo models because it is fast. This section calls function f_E2E_pedigree_08032018 to generate uncertainty terms for all elements of the ECOTRAN EnergyBudget_matrix (A_{cp}) as Coefficients of Variation (CV). Function f_TerminalDetritus_08032018 is called to generate an ULTIMATE detritus pool where all unconsumed terminal benthic detritus flows and is subject to bacterial metabolism. This function is necessary for the management of bacterial recycling of benthic detritus in time-dynamic simulations. Function f_E2E_MonteCarlo_08032018 generates a set of randomly generated ECOTRAN models, multiple alternate versions of the EnergyBudget_matrix (A_{cp}). (see section "3: Generate Monte Carlo models").

Step 5 - Define the transfer efficiencies for each group. Because the EnergyBudget_matrix defines the fate of all consumption (rather than only of production), we define the transfer efficiencies to all be 1 EXCEPT for the pelagic ammonium pool (TE_{NH4}). We have adjusted TE_{NH4} for food webs run within each physical setting so that model runs have appropriate f-ratios under reflective ocean boundary conditions.

Step 6 - Define the physical geometry of the model and prepare physical advection, mixing, and sinking rate time-series.

There is a function specific to each physical setting called at this step. Use ECOTRANdynamic_context_08032018 with upwelling, downwelling, or bank physics:

f_ECOTRANphysics_upwelling_08022018	upwelling
f_ECOTRANphysics_downwelling_08022018	downwelling
f_ECOTRANphysics_bank_08022018	bank

Use ECOTRANdynamic_context_basin 08032018 with basin physics:

f_ECOTRANphysics_basin_08022018	basin

Step 7 - Prepare Production_input driver time-series that drives the ecosystem. This is the external input from offshore and/or alongshore and/or river that enters the model domain via advection and mixing. Under the reflective boundary assumption, only specified groups (NO_3) are advected into the model domain, and the code eliminates advection of non-NO_3 groups (or of empty water) into the domain. The reflective boundary assumption is that biomass of non-NO_3 groups is the same on either side of model domain outer boundaries.

Step 8 - Prepare non-predation BioenergeticBudget terms. These are weight-specific growth & consumption rates for each group, and they can vary with time if wanted.

NOTES:

- Deactivated sub-steps are for use with Michaelis-Menten nutrient uptake kinetics and/or linking with NPZD plankton models.

Step 9 (deactivated) - Define functional predator-prey relations.

NOTES:

- Functional predator-prey relations are defined later within Monte Carlo looping step (sub-step 11c).

Step 10 - Define Michaelis-Menten functional predator-prey relations for primary producers.

NOTES:

- Michaelis-Menten mechanics are not required and are commented-out when not used

Step 11 - Cycle through each Monte Carlo model by selecting one ECOTRAN EnergyBudget_matrix (A_{cp}) at a time.

sub-step 11a - initialize solution time variable.

sub-step 11b - begin looping through each Monte Carlo model.

sub-step 11c - set functional response terms for the current Monte Carlo model. This step calls function f_FunctionalResponse_MonteCarlo_09122016 to prepare array of vulnerability terms and allows for random generation of functional response terms within a predefined uncertainty level.

NOTES:

- at present, all functional response terms are forced to linear for special comparative shelve study

sub-step 11d - select the current Monte Carlo model.

sub-step 11e - set food web models for each sub-region. There are two options: use the same food web for all sub-regions or use different food webs for each sub-region.

NOTES:

- at present, all sub-regions are set to use option 1: same food web model used in all sub-regions
- special case difference for basin setting in code ECOTRANdynamic_context_basin 08032018

sub-step 11f - calculate sinking fluxes as the fraction of consumption inflow to a group sinking OUT of the source box.

sub-step 11g - pack variables into ODE structure variable

sub-step 11h - define initial conditions as biomass (nitrogen) flowing into each model group. There are two options:

METHOD 1: use for driving initial model conditions with primary production defined by Michaelis-Menton uptake. Calls function f_MichaelisMenten_05152016.

METHOD 2: use for driving initial model conditions with primary production defined by ECOPATH mean condition parameterization, P = [(P/B) * B]

Both methods call function f_InitialProductionRates_05112016.

sub-step 10i - calculate BOUNDARY biomass conditions. There are three options:

OPTION 1: ocean biomass conditions at time t are estimated from outer-shelf sub-domain food webs as driven by NO3 input time-series. Calls function f_StaticProductionTimeseries_09042017. (NOTE: use with ODE function f_ECOTRANode_DefinedBoundary_08032017 or f_ECOTRANode_DefinedBoundary_basin_08032017)

OPTION 2: empty ocean (ocean only has NO3). This option is used just for de-bugging code. (NOTE: use with OPTION 1 activated and with ODE function f_ECOTRANode_DefinedBoundary_08032017 or f_ECOTRANode_DefinedBoundary_basin_08032017)

OPTION 3: reflective boundary assumption (used in comparative shelves study). (NOTE: use f_ECOTRANode_ReflectiveBoundary_05182017 or f_ECOTRANode_ReflectiveBoundary_basin_05242017)

sub-step 11j - pack more variables needed for ODE solver.

sub-step 11k - solve the dynamic model using MATLAB ODE solver ode23t.

OPTION 1: for defined ocean boundary biomass conditions (step 11i - OPTION 1 & OPTION 2), use f_ECOTRANode_DefinedBoundary_08032017 for upwelling, downwelling, or bank settings and use f_ECOTRANode_DefinedBoundary_basin_08032017 for the basin setting

OPTION 2: for reflective ocean boundary conditions (step 11i - OPTION 3; used in comparative shelves study), use use f_ECOTRANode_ReflectiveBoundary_05182017 for upwelling, downwelling, or bank settings and use f_ECOTRANode_ReflectiveBoundary_basin_05242017 for the basin setting

NOTES:

- trial-and-error suggests ode23t has a bit better performance than using ODE45
- when using

sub-step 11L - unstack result (store_ProductionRates) to retrieve spatial boxes

Step 12 - save results and work loop to next Monte Carlo model.

7) references

Chiaverano LM, Robinson, KL, Tam, J, Ruzicka JJ, Quiñones, J, Aleksa KT, Hernandez FJ, Brodeur RD, Leaf R, Uye S, Decker MB, Acha M, Mianzan HW, Graham WM (2018) Evaluating the role of large jellyfish and forage fishes as energy pathways, and their interplay with fisheries, in the Northern Humboldt Current System. Progress in Oceanography 164:28-36.

Collie, J.S., Gifford, D.J., Steele, J.H. (2009) End-to-end foodweb control of fish production on Georges Bank. ICES Journal of Marine Science. 66:2223-2232.

de Haast, J.A., Treasure, A.M. , Ruzicka, J. J., Moloney, C.L. (2017) A donor-driven approach to modelling anchovy-sardine dominance shifts in the southern Benguela ecosystem. Journal of Marine Systems. http://dx.doi.org/10.1016/j.jmarsys.2017.09.001

Robinson, K.L., Ruzicka, J.J., Decker, M.B., Brodeur, R.D., Hernandez, F.J., Quiñones, J., Acha, E.M., Uye. S.-I., Mianzan, H., Graham, W.M. (2014) Jellyfish, forage fish, and the world's major fisheries. Oceanography 27:78-89.

Robinson, K.L., Ruzicka, J.J., Hernandez, F.J., Graham, W.M., Decker, M.B., Brodeur, R.D., Sutor, M. (2015) Evaluating energy flows through jellyfish and gulf menhaden (Brevoortia patronus) and the effects of fishing on the northern Gulf of Mexico ecosystem. ICES Journal of Marine Science. DOI: 10.1093/icesjms/fsv088.

Ruzicka, J. J., K. H. Brink, K.H., Gifford, D.J., Bahr, F. (2016) A physically coupled end-to-end model platform for coastal ecosystems: Simulating the effects of climate change and changing upwelling characteristics on the Northern California Current ecosystem. Ecological Modelling 331:86-99.

Ruzicka, J.J., Brodeur, R.D., Emmett, R.L., Steele, J.H., Zamon, J.H., Morgan, C.A., Thomas, A.C., Wainwright, T.C. (2012) Interannual variability in the Northern California Current food web structure: changes in energy flow pathways and an End-to-End model to investigate system response to alternate forcing scenarios. Progress in Oceanography 102:19-41.

Ruzicka, J.J., E.A. Daly, Brodeur, R.D. (2016) Evidence that summer jellyfish blooms impact Pacific Northwest salmon production. Ecosphere 7(4):e01324. 10.1002/ecs2.1324.

Ruzicka, J.J., Steele, J.H., Ballerini, T., Gaichas, S.K., Ainley, D.G. (2013) Dividing up the pie: Whales, fish, and humans as competitors. Progress in Oceanography 116:207-219.

Ruzicka, J.J., Steele, J.H., Brink, K.H., Gifford, D.J., Bahr, F. (in press August 2018). Understanding large-scale energy flows through end-to-end shelf ecosystems - the importance of physical context. Journal of Marine Systems.

Ruzicka, J.J., Steele, J.H., Gaichas, S.K., Ballerini, T., Gifford, D., Brodeur, R.D., Hofmann, E.E. (2013) Analysis of energy flow in US GLOBEC ecosystems using end-to-end models. Oceanography 26:24-39.

Steele, J.H. (2009) Assessment of some linear food web methods. Journal of Marine Systems 76:186–194.

Steele, J.H., Ruzicka, J.J. (2011) Constructing end-to-end models using ECOPATH data. Journal of Marine Systems 87:227-238.

Treasure, A., Ruzicka, J., Moloney, C., Gurney, L., Ansorge, I. (2015) Land-sea interactions and consequences for sub-Antarctic marine food webs. Ecosystems. 18:752-768.