NeutralCom – a FORTRAN program for generating neutral macroecological models

Version 1.2

Werner Ulrich

Nicolaus Copernicus University in Toruń

Department of Animal Ecology

Gagarina 9, 87-100 Toruń; Poland

e-mail: ulrichw @ uni.torun.pl

Latest update: 30.06.2005

1.1. Introduction

Recently, neutral macroecological models have come into the focus of ecological interest. Although the application of neutral models has already a long tradition in genetics and evolutionary theory (Kimura 1985, Kimura et al. 1994) and were proposed by Caswell in 1976 for explaining basic patterns of community structure in ecology they received only few attention of the ecologists of that days. This little interest is surely explained by the fact that the dominating paradigm of that days was that biotic interactions especially competition (even as a ghost of competition past) or predation (in the widest sense) shape local communities. Additionally, the focus of interest was the ecosystem with its local communities or guilds. This view changed with the rise of Macroecology, the ecological branch that deals with patterns above local scales, but also with the rising number of studies that denied the importance of competition and trophic relations.

This paradigm change resulted at last in a renewed interest in neutral models. Especially the models proposed by Stephen Hubbell (1997, 2001) and Graham Bell (2000, 2001) inspired ecologists and caused intensive work in the field (Whitfield 2002).

What are neutral ecological models? Shortly speaking these are models that treat species not as biological entities that interact but treat them as lottery balls without biotic interactions. Ecological, especially macroecological, patterns are then generated from pure chance processes. Appealing are these models for ecologists because Hubbell and Bell were able to show that many ecological patterns that were earlier explained by biotic interactions can be generated from neutral models with a very limited number of parameters. The most parsimonious versions require only birth and death rates and assume local saturation of individuals irrespective of species. Mathematically speaking, the neutral models of Bell and Hubbell treat local communities as zero sum multinomials driven by birth and death processes and allow for local immigration and emigration to held the local individual number constant.

The present program is an extension of the class of models Graham Bell (2001) introduced. Starting point of these models is a metacommunity of S species and N individuals. These

metacommunity is divided into P local communities each of S_i species and N_i individuals. The original version of Bell's model assumed local saturation with individuals and fixed individuals numbers. The model starts from a metacommunity in which all species have identical individual numbers. In each step each local community loses d randomly chosen individuals (they die) irrespective of species and gain b randomly chosen ones (they are born). If d > b d-b individuals are allowed to immigrate from the metacommunity, if d < b b-d individuals emigrate. Local abundance (the total number of individuals) and metacommunity abundance are held constant.

The model of Hubbell instead includes also random speciation events allowing therefore for a variation in total metacommunity species numbers. It is therefore also an evolutionary model. *NeutralCom* generates new species from a speciation probability parameter that tells how probable a certain individual (irrespective of species) mutates into a new species. The program models therefore a point mutation mode. The actual version of the model does not trace lineages allowing evolutionary pathways being studied.

Bell (2001) and Hubbell (2001) were able to show that such a simple process is able to generate a series of basic ecological patterns with realistic parameter values. These include

- Sigmoid shaped relative abundance distributions (frequency—rank order distributions) (Hubbell 2001)
- Allometric species—area relations (Hubbell (2001)
- Exponential abundance—range size relations (Hubbell 2001)
- Similarity—distance relations (Condit et al. 2000, 2002).
- Fractal and self-similar phylogenetic clade patterns (Hubbell 2001)
- A fractal nature of biodiversity (Hubbell 2001)

1.2 *NeutralCom* extensions of previous models

The models of Hubbell and Bell have a series of shortcomings. They do not deal with body weight and community biomass related ecological patterns. These include especially density—body weight distributions (Gaston 1993, Currie and Fritz 1993, Cyr et al. 1997), biomass—species diversity relations (Ulrich 2003), range size—body weight

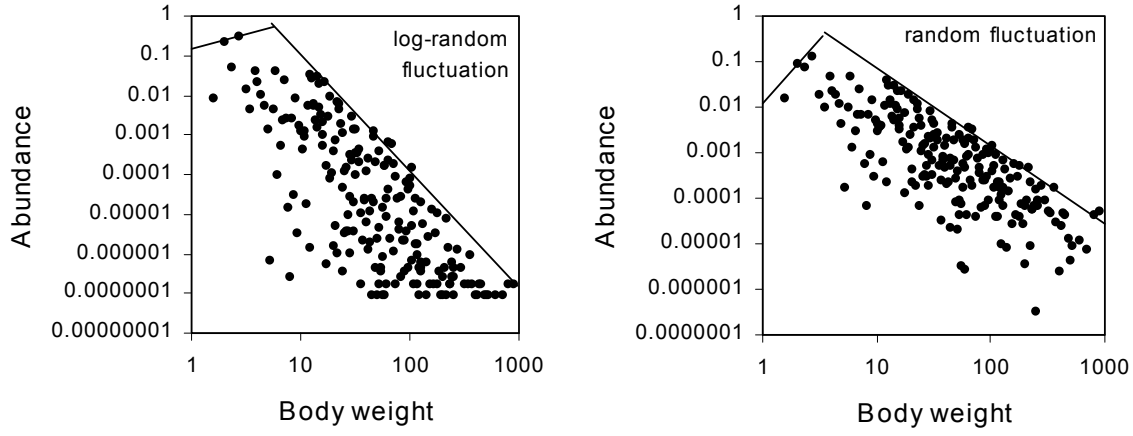

Figure 1: Abundance – weight distributions (AWD) of 2 model communities with 200 species each and a SWD with a median of 8 and a variance of 2. A: abundances were assigned with random numbers on a logarithmic scale; B: abundances resulted from random numbers on a linear scale.

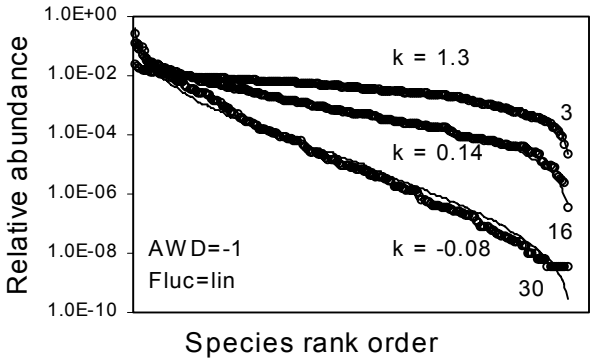

Figure 2: Abundance—weight distributions of three model assemblages generated by the three step method described in the text. We see that typical RADs appear that may be fitted by common models of relative abundance. In this case the power fraction model of Tokeshi (1996) was used for fitting (using the program RAD, Ulrich 2002). The *k*-values refer to the shaping parameter of this model. A power fraction model with $k = 0.1$ would be similar to a canonical log-normal, $k = 1.0$ is nearly identical to a broken stick model. The AWD has a slope of –1, linear random numbers were used to assign densities and model species body weights of the assemblages span over 3 to 30 binary weight classes.

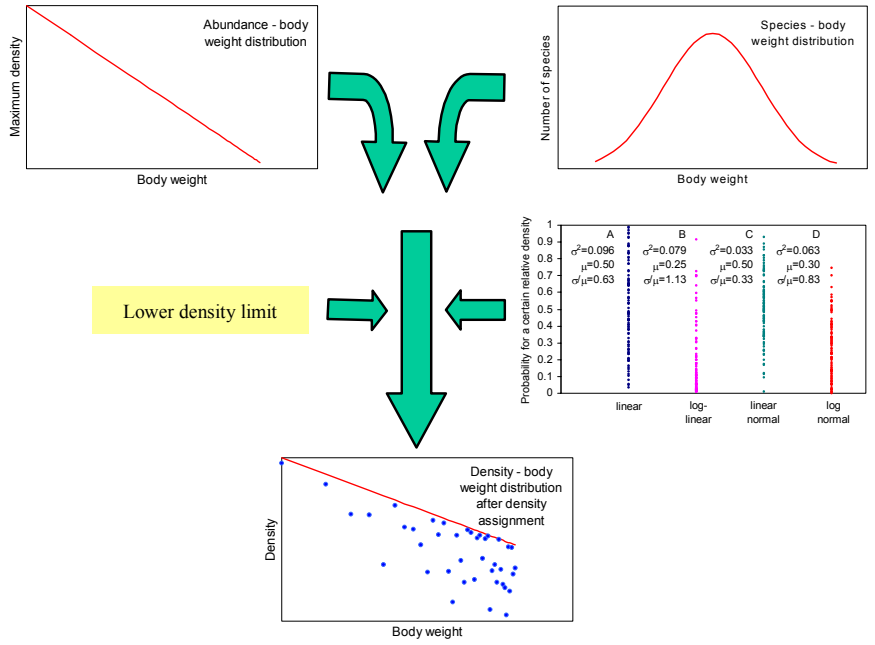

Figure 3: A three step process generates density—weight distributions and also relative abundance distributions. In this process two basic body weight dependent ecological distributions work together. In a first step a density—weight distribution is defined. A second step then assigns species from a given species—weight distributions hypothetical body weights. Random density fluctuations of these species together with an also given lower density limit result then in realistic RADs and DWDs.

relations (Ziv 2000), and species diversity—productivity relations (Gaston 2000), and the study of succession.

Additionally, they are static models insofar as they assume total individual numbers (at the local and the metacommunity scale) being fixed. At last, they do not include heterogeneity of habitat structures or resource distribution.

The present program extends the models of Hubbell and Bell to include species body weights and habitat heterogeneity. In doing so it uses the patch occupancy module of *PatchOccupancy*, a program designed by me do model classes of patch occupancy models (Ulrich 2002a). This program places individuals of species into the cells of a grid by a three step process. In a first step a species—body weight distribution is generated, the second step assigns species densities, and the third step places the individuals into the cells, where the cell properties can be varied to simulate a wide variety of heterogeneity patterns.

Then the presents program models death, birth, immigration, emigration, and speciation patterns according to the models of Bell and Hubbell. At last, *NeutralCom* returns again to the *PatchOccupancy* kernel and samples the resulting grid according to the wide possibilities this program offers.

Table 1: The batch file *Neutral.txt* generated by *NeutralCom*. The file contains all the necessary parameter values to run the program. Other filenames are also allowed, the input needs not to be formatted only the sequence of variables has to be kept. The number of input lines is unlimited.

Areal	Sam p A	Max-Sa	minSa	Step Sa	Ind-Fac	Met-hod	Spe-cies	maxD	minD	Dfluc	Fluc-my	Fluc-StD	DWD	SWD	Class	SWD my	SWD-StD	DWD my	DWD StD	Aggreg	Area-Het	Rnd-Het	Rnd-FracM	Sca-le1	Sca-le2	Birth	Migra	Spec-Rate	Div-Ind	DFix	Grid-Het	Spec M	Suc c	Disp
100	20	12	4	4	1	dep	20	10	1E-08	0	0.5	0.01	power	normal	6	3	1	0.9	0	2	5	md	rmd	2	1.5	0.014	0.014	1E-07	sha	0.2	fix	r	succ	o
1000	1	1	1	1	10	ind	1	0.1	0.001	0	0	0	linear	logn	1	1	1	1	0	1	10	rnd	rmd	2.3	1.3	0.001	0.1	1E-08	simp	0	rnd	p	ma-in	1.0
200	100	100	100	1	100	all	200	10	0	0	0.5	0.01	rnd	logn	30	5	2	1	1	0.5	20	fix	super	1.5	3	0.02	0.001	1	evar	0.2	rnd	p	succ	0.1
20	20	100	100	1	20	oct	100	1	0	0	0.5	0.01	nor-mal	nor-mal	10	5	2	10	1	3	20	fix	super	2	3	0.1	0.001	5E-08	sha	0.9	fix	r	no	0.01
100	100	100	100	1	100	seq	100	1	0	0	0.5	0.01	logn	nor-mal	10	5	1	1	0.1	3	20	fix	rmd	2	3	0.001	0.001	1E-08	evar	1	fix	r	no	1.0

2. Generating communities with the *PatchOccupancy* module

Model species assemblages are generated using underlying relative abundance relationships (often log-normal or log-series models, May 1975, Leitner and Rosenzweig 1997, Wilson et al. 1998, Ulrich 2002b). The starting point of Hubbell is a log-series. The Bell class of models however starts from a simple random distribution of abundances in the metacommunity.

Real species assemblages are characterized by three basic features. They have a certain species - weight distribution (SWD) [nearly always normal or log-normal (Novotny and Kindlmann 1996)] and an abundance - weight distribution (AWD; often also termed density—weight distribution DWD) with an upper boundary line (most often described by a power function), which has frequently - but not always - a triangular shape (Currie & Fritz 1993, Currie 1993, Ulrich 1999c). Thirdly, species do not have infinitely low densities, there is a lower density limit below which a species goes locally extinct.

Therefore, *NeutralCom* first generates model communities (assemblages) from these three features. Firstly, a SWD is generated from three based input variables: the number of logarithmic body weight classes, the modal body weight class and the variance. SWDs can be defined as normal (input *normal*) or as lognormal (input *logn*).

In a next step a DWD has to be defined. Typical DWDs are power functions (input *power*) with a negative slope (Ulrich 1999c). *NeutralCom*, however, allows also other distributions to be computed: a linear (input *linear*), a normal (*normal*), a lognormal (*logn*), or a random (*rnd*). The DWD has then to be defined by a slope (in the *power* or *linear* case) or by a mode and a variance (Fig. 1). To give all species initially equal densities a linear DWD with slope and variance zero have to be defined. A variance larger than zero assigns initial densities from simple linear random numbers. Densities from random numbers at a log-scale are assigned from a power function DWD with slope 0 and a variance larger than zero.

In reality species with very high and very low densities are less often found than ones with intermediate abundances. Sometimes, low densities are more probable than high ones. To simulate such a pattern densities are either assigned by random fluctuations with log-transformed densities (resulting in a higher number of low density species) or using normally distributed random numbers (resulting in a higher number of intermediate densities). Additionally, simple linear random numbers can be used to assign densities. An example of the resulting relative abundance distributions con-

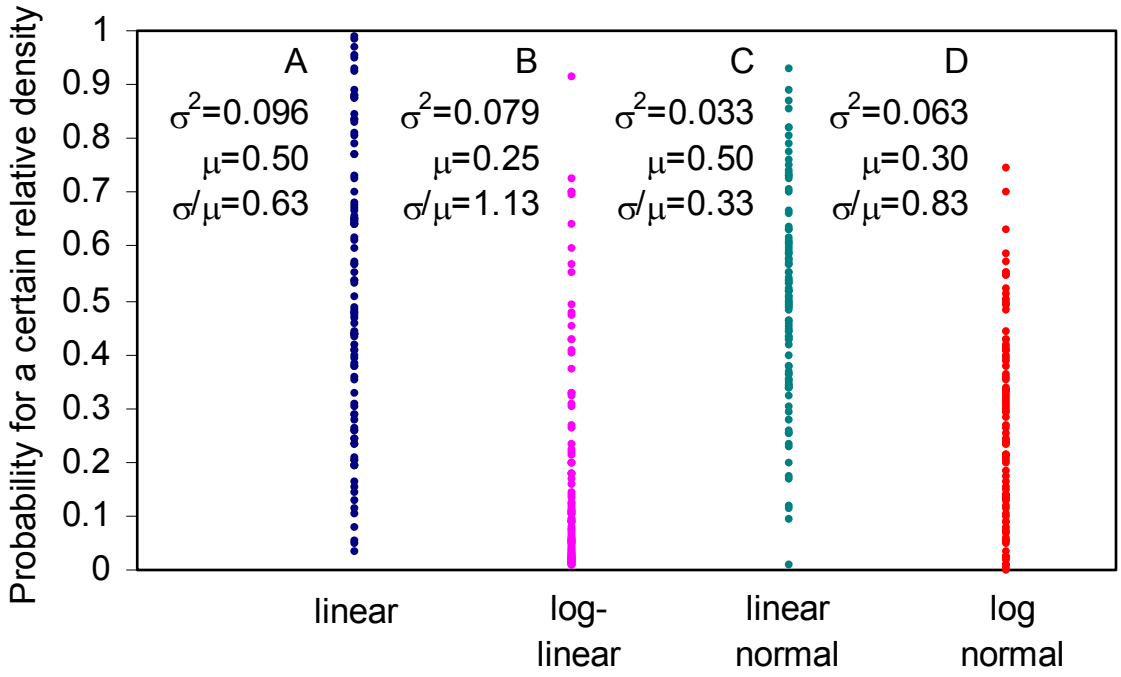

Figure 4: Four possibilities how densities may be assigned. In case A variability is a linear random variable, that means all densities between an upper and a lower boundary are equally probable; In B log transformed random number are used that lead to a lower mean density; C and D equal A and B but normal random numbers are used leading to a situation that intermediate densities (either on a linear or on a log scale) are more probable. Given are in each case the associated mean m, the variance s2 and the coefficient of variation CV (s / m) as a measure of variability. Note that CV is always higher after log transformations. Normal distributed densities around the mean reduce the value of CV.

tains Fig. 2. This Figure also shows that the assemblages have typical density-weight distributions (Currie 1993, Ulrich 2001). The main idea behind the whole process of assigning species densities is again exemplified in Fig. 3.

However, it is also possible to run *NeutralCom* either with real data about relative abundances and species body weights or with theoretical distributions generated by other programs like RAD (Ulrich 2002b). By this way *NeutralCom* predicts future cell abundances and therefore patch occupancies. It may be seen as an extension of existing software for generating Markov chains (Hubbell 2001). In fact, every step of a neutral model is a special form of a Markov chain were transition probabilities result from the five basic model parameters (birth, death, immigration, emigration, and speciation rates). Future versions of the program will develop this feature iin more detail and will explicitly compute transition probability matrices. It will also be able to read whole species x sites matrices .

3. Running the program

3.1. Computing a model assemblage

After program start *NeutralCom* first asks 'What to do?'. You can either model and sample a predefined assemblage from the cells of a grid (see below) (input: *s*), or printing the basic batch file *Neutral.txt* (Tab. 1) with which you can run *NeutralCom* (input: *b*).

After the input s the program asks whether you want to specify the output filenames by adding endings to them. This option allows multiple program run simultaneously. A simple *Enter* retains the output filenames.

Then the program asks whether you want to print the surface structure of the grid (see below).

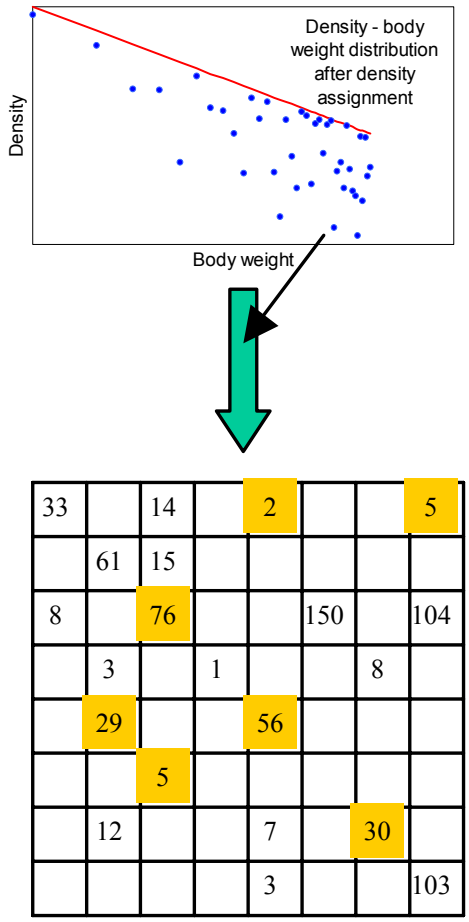

Figure 5: Sampling cells of a grid of 64 cells. The individuals of the community defined in Figure 2 are spread onto a grid of 64 cells. This is shown for one highly aggregated species (marked by the arrow). Afterwards 7 cells (yellow) are sampled.

If you answer *yes NeutralCom* will print the mean individual numbers per species that can be placed into the cells per placing step (see below). The respective output file is *Grid.txt*. Essentially, the values printed reflect the grid surface structure.

After the model option *NeutralCom* asks whether you want to run the program by hand or via a batch file. Table 1 shows an example of this batch file that is printed by the program. Batch files allow multiple modelling during one program run.

To compute a model the following input options are needed:

Area and SampA: Define by this option a hypothetical area (the grid) in which the species occur. An input of a *Area*-value i defines the area i * i cells, an input of 10 defines therefore a grid of 100 cells. Sometimes it is convenient to sample not the whole grid but a predefined part of it. With the option *SampA* you define such a part (located randomly inside the total grid). For instance, if you defined the size of the grid with *Area* = 1000 you can define an area to be sampled of S*ampA* = 10. The effect is that only from the cells from i =*ran* to *ran* + 10 and from j = *ran* to *ran* + 10 samples will be taken

Species: Give the number of species of the assemblage. The maximum number is 1000.

maxD: Here, you define the mean density per cells p_1 (this is the input value) of the most abundant species. The total

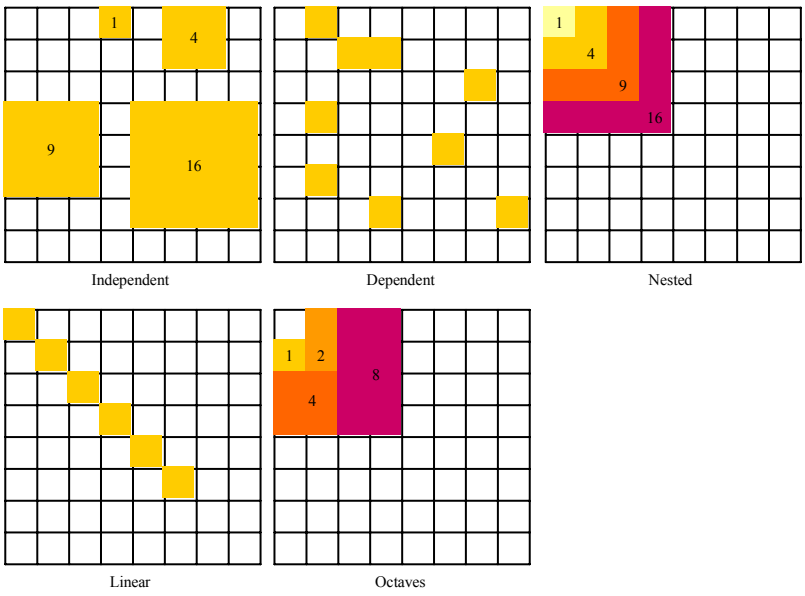

Figure 6: Fives types of sampling cells from a grid. Independent: In this case adjacent quadratic areas of different largeness are taken. The specification *independent* stems from the way the program computes species—area relationships. In this case SARs are computed from independent areas. Dependent: a series of basic units of sampling (most often single cells) are sampled. SARs are computed from random combinations of these cells (see below), therefore the classification dependent. Nested sampling starts from one corner and proceeds sequentially. Linear: samples one of the diagonals of the grid. Octaves: samples contingent independent areas of which the next larges is always two times as large as the one before. In this latter case the last sample is always the whole square in which the samples are (in this case the sum of samples with areas 1, 2, 4, and 8 plus the blank cell.

number of individuals of that species is calculated as *maxD* = p_1*i*i. If you set *maxD* to zero densities are assigned so that the least abundant species will have exactly one individual.

minD: Sets the mean density of the least abundant species. Together the above two values define the range of densities of the community. The input 0 sets the minimum density to 1 individual in the whole grid (*minD* = 1/(i*i).

DWD: Here you define the type of the density—weight distribution (or abundance—weight distribution) that defines the upper range of possible densities for each species. *NeutralCom* can assign densities using five different types of models. Most often used (and most often found in nature) is a power function type of DWD (input: *power*). In this case the DWD is assumed to follow the a power function: D =*maxD* * W^{-z}. You may also use a linear model (D = *maxD* – zW; input: *linear*), a normal model: D=*maxD* * exp(-(z - mean)^2/2var); input: *normal*), a log-normal model (D=*maxD* * exp(-(ln(z) - mean)^2/2var); input: *logn*), or a random model (D=*maxD*).

The next two input variables define now the parameter values *z* and *var*.

DWDmy: This is the input value for *z*.

DWDStD: This is the input value for the standard deviation of the normal and log-normal model (*var*=*StD*^{2}).

Dfluc defines the type of random numbers by which densities are assigned. The input 0 lets the program using random numbers on a linear scale, the input 1 random numbers on a logarithmic scale and an input i between 0 and 1 causes the program to use for i% of species a logarithmic and for (100-i)% a linear scale.

Flucmy: Random numbers used for density assignment may be linearly or normally distributed. The input 0 causes linear distributed random numbers, an input between 0 and 1 defines the mean of a normal distribution.

FlucStD: the input 0 defines again linear random numbers. An input between 0 and 1 defines the standard deviation of the normal distribution. The above three input parameters define therefore four different types of randomness by which densities might be assigned: linear and normal distributed random numbers either on a linear or on a logarithmic scale (Fig. 4).

The next four input variables define the species— weight distribution:

SWD defines the type of the species—weight distribution. You may either choose a normal distribution (over $\log_2$ body weight classes)(input: *normal*) or a log-normal distribution (input: *logn*). The species defined in *Species* are assigned hypothetical body weights according to the distribution below defined by the three parameters *Class*, *SWDmy*, and *SWDStD*.

Class defines the number of $\log_2$ body weight classes.

SWDmy defines the mean of the SWD.

SWDStD defines the standard deviation of the SWD. The species defined in *Species* are now assigned hypothetical body weights according to the above defined distribution.

The above parameters fully define a model assemblage of S species and result in species rank order distributions of which three examples are shown in Fig. 2.

3.2. The model run

NeutralCom proceeds in a stepwise manner. In a first step it adds a number *b* of individuals (irrespective of species) to the first cell of the grid as defined by the parameter *birth* (*b* = (*ran* * *birth* + *birth)* * *density*) where *ran* is a normal random number with mean 0 and variance 1. Density is the total number of individuals in that cell irrespective of species. In a second step *b* individuals (irrespective of

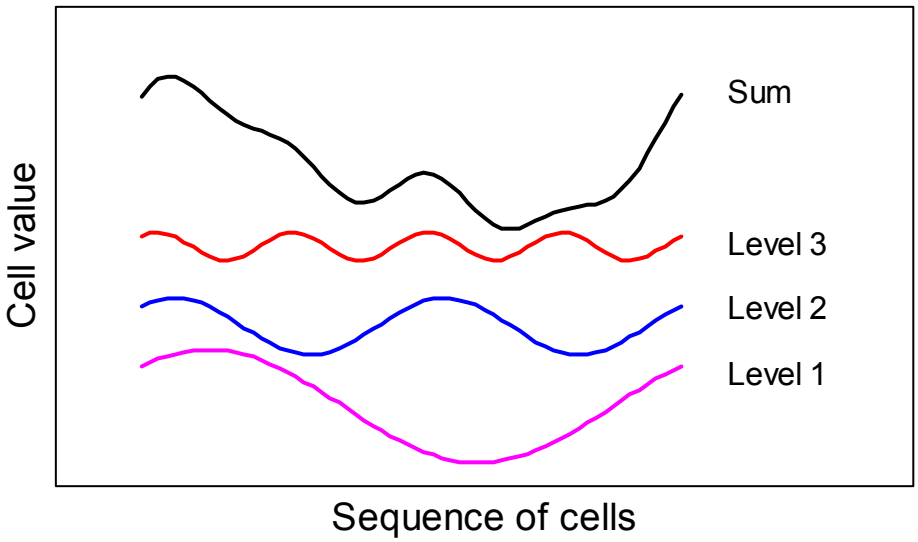

Figure 8: The superposition of trigonometric functions [in this case sin(x)=cos(x)] results in an irregular pattern of cell properties that that can be described by fractal geometry (for further details see Burrough 1983).

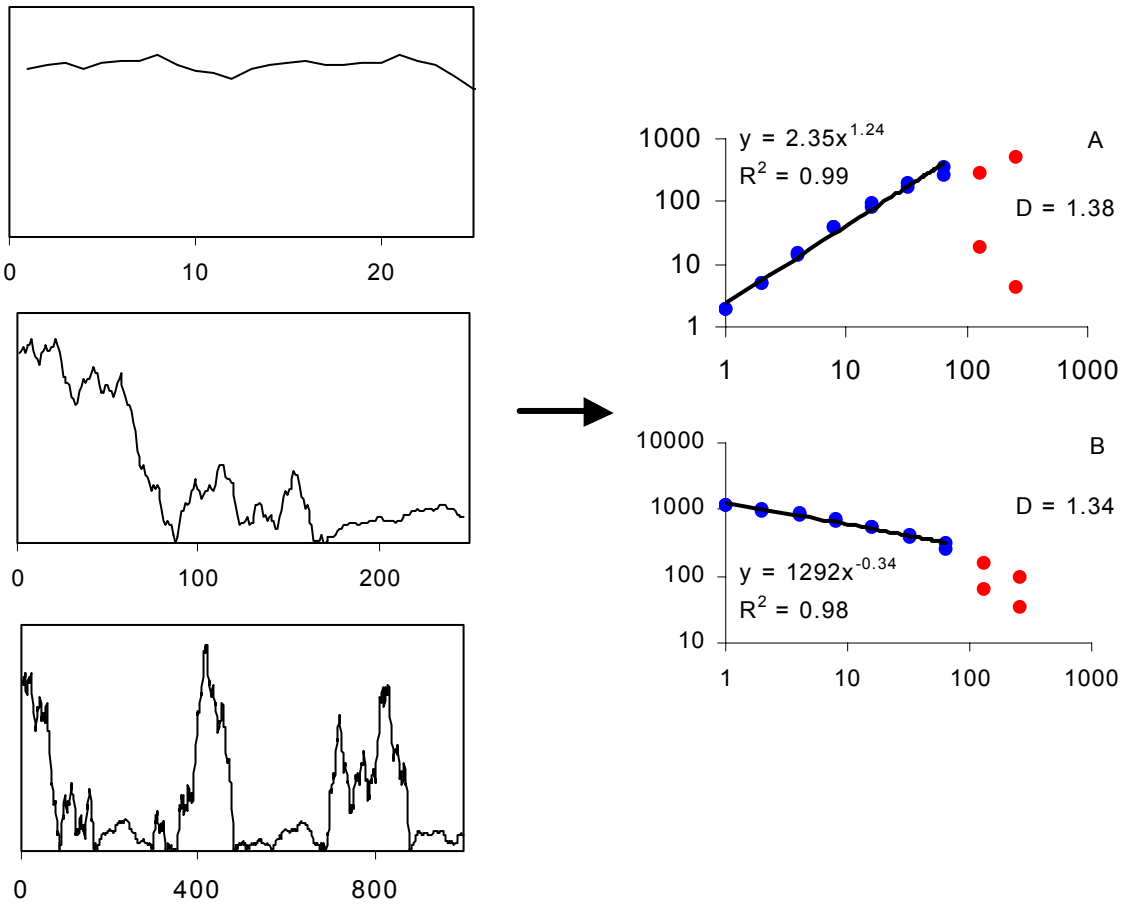

Figure 9: The diagonal of a grid (1000x1000 cells) with parameter settings that are shown in Table (*AreaHet* = 20, *RndHet* = rnd, *Scale1* = 2.2, *Scale2* = 2.7) seen under three different magnifications. From the two grid diagonals fractal dimensions are computed using either the semivariogram method (A) or the ruler length method. Note that the red data points already belong to a different scaling region. *NeutralCom* excluded them from computation.

species) are eliminated at random.

Then, the program either allows for emigration or for immigration defined by the parameter *MigRa* that give the fraction to individuals that will migrate. Single individuals immigrate from or emigrate to neighbouring cells by a chance process where the chance to come from or to go to a cell is distance dependent. Nearer cells have a greater chance to be affected. An Gaussian model of the form

$$p = Norm * disp$$

is used as the respective probability function where *Norm* is a normally distributed random number and disp the dispersal parameter. Target or source cells are then fulfilled by a death or birth process as before.

Either the whole process is repeated until all cells have their initial species number (after the input value (*Dfix*=0). For values of *DFix* larger than zero this value defines the standard deviation of a normally distributed random number (*ran*) around the initial number of individuals of the actual cell (*celldensity*). This standard deviation is given by *ran*Dfix*celldensity*. *NeutralCom* computes in this case after the birth and death process a new total density for the actual cell (but not for the above target or source cells for emigration and immigration).

The whole process is step by step repeated for all cells of the grid. Therefore, larger grids with many species require extraordinary computing times.

After each step the program adds new species according to the speciation probability defined by *SpecRate*. It is the probability of a speciation event per number of birth processes. *SpecRate* is a very sensitive parameter. Too low or too high rates immediately lead to unrealistic low or high speciation events.

NeutralCom allows for two different modes of speciation, the point mutation mode (input *SpecM* = *p*) and the peripheral isolate mode (input *SpecM* = *r*).

As shown by Hubbell (2001) under the point mutation mode are speciation probability and total metacommunity size N related by

$$SpecRate = \frac{\Theta}{2N}$$

with θ being the fundamental biodiversity number that approximates in the long run and for large N the α-diversity index. Therefore *SpecRate* should in most cases be adjusted to values that result in θ values between 1 and 1000. In the long run *SpecRate* and N fully define local and metacommunity structure.

If you define *SpecRate* = 1 or larger the program keeps the initial species number constant adding after each step as many species as had been lost.

With the parameter *IndFac* you define how often the whole process is repeated. *IndFac* is a real values with which the total number of individuals in the grid is multiplied. For instance, if you have 10000 individuals in the grid and define *IndFac* as 10 the whole process is repeated until 100000 individuals were born, died, immigrated, or emigrated.

3.3. Studying succession

NeutralCom allows two types of succession to be studied (input: *Succ* = *succ, main,* or *no*). After the initial placing of all individuals into the

* Density	Body weight (micoGr) Community A
388	39.9
229	7.1
85	164
37	548.1
* Density	Body weight (micoGr) Community B
0.57	5.7
0.355	138.6
0.28	7.9
0.25	163
0.15	98.4
0.1	163
0.055	391.6

Table 2: An input file with data from two communities. The first column has to contain abundance data, the second can (but need not) contain species body weights. The file needs not to be formatted

Table 3: The output file *Distribution.txt* that shows the original community used or generated by the program.

Characteristics of whole community

Areal	Samp	maxSa	minSa	Step-	IndFac	Met-	Spe-	maxD	minD	Dfluc	Flucm	Flu-	DivInd	Dfix	Model	Const	Const	TruncF
5	3	9	9	1	10	all	20	2	0.04	0	0	0	sha	0		0	0	0
DWD	SWD	Class	SWDmy	SW-Dvar	DWDmy	DWDvar	Aggr	Area-Het	FracM	Scale1	Scale2	Birth	Death	Spec-Rate	Rnd-Het	Grid-Het		
linear	nor-mal	30	15	2	0	0	0	0	rmd	2	1.5	0.1	0.1	0.2	rnd	fix		

No. of species per weight class

1	1	2	2	2	3	2	2	2	1	1

SpeNo	We-ight	Org.-Ind	Ass-Ind	Density boundary	Genus	Family
1	2	50	50	50	2	3
2	4	50	50	50	3	1
3	6	50	50	50	1	2
4	8	50	50	50	5	3
5	12	50	50	50	6	4
6	16	50	50	50	1	1
7	24	50	50	50	1	2
8	32	50	50	50	3	3
9	42.67	50	50	50	3	1
10	53.33	50	50	50	5	2
11	64	50	50	50	7	1
12	96	50	50	50	2	3
13	128	50	50	50	3	2
14	192	50	50	50	2	1
15	256	50	50	50	2	1
16	384	50	50	50	1	1
17	512	50	50	50	6	1
18	1024	50	50	50	8	4
19	2048	50	50	50	9	2
20	0	0	0	0	4	1

No. spe-cies	Ind. of Ass.	Total biomass	SD lb (dens)	RAD slope	RAD Inter	Diver-sity	Even-ness	Alpha	Theta
19	950	245200	0	0	-2.944	2.944	1	3.37	380

grid the program clears one randomly chosen cell (*'succ'*) or clears the whole grid excpet the cell (1,1) ('main') from which individuals now immigrate into the whole grid. By this way *NeutralCom* simulates a mainland island pattern. In the 'succ' model the program will clear an additional cell after half of the program run . For the cleared cells no zero sum game is played for immigration processes until the density in this cell reaches the initial density before clearing. In the case of the island mainland pattern a third input file can contain the desired individuals numbers of the later matrix. Note that the cell (1,1) serves in thjs case as a controler. The number given there is a multiplier for the total number of individuals defined in the file of the relative abundances or computed by the program. By this way you defined the total number of individuals in the later grid. Birth, death and emigration events are modelled as before. In the case of succession, the total individual number in the grid is

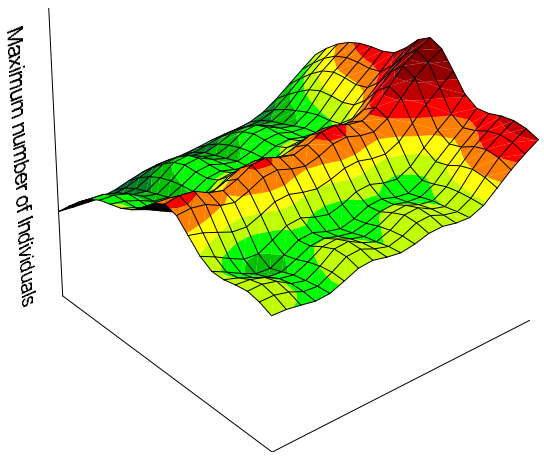

Figure 7: Cells of a grid with 20 * 20 cells may differ in the maximum number of individuals per species that can be placed in each step.

therefore at the end higher than at the beginning. The difference is approximately the mean cell density in the grid.

3.4. Sampling individuals from a grid

If you compute a model assemblage as described above or you took data from a real community (see below) you can spread the individuals of this community into a grid defined by *Area*. Then, you can take samples (sample units are most often whole cells of this grid or combinations of cells) and analyse the resulting patterns. *NeutralCom* offers a wide range of options to define the cell properties as well as the way of sampling.

3.4.1 Defining the sample process

To define the sample process five input parameters are necessary. The program samples whole sets of cells during one program run. So, you can define a whole sampling program at once.

maxSa defines the maximum number of sample units to be sampled

minSa defines the minimum number of sample units to be sampled

StepSa defines the step width in the number of sample units to be sampled. For instance after the definition of *maxSa*=100, *minSa*=10 and *StepSa*=10 *NeutralCom* samples during one program run 10, 20, 30, 40,… 100 cells and analyses the resulting patterns independently for each step (for 10 cells, then for 20, for 30, and so on).

Method: *NeutralCom* offers five basic methods of sampling the grid (Fig. 6): a nested design (input *seq*), an independent design (input: *ind*), and a dependent design (input: *dep*), a linear design (*lin*), and sampling of independent octaves (*oct*). In the cases of *seq* and *ind* maximum sample sizes (defined by *maxSa*) are in cases of too high input values adjusted to maximum possible sample sizes. Additionally, it is possible to define *Method=all*. In this case the program prints the whole sample area.

3.4.2. Species spatial distribution patterns

Species often occur in an aggregated manner. *NeutralCom* simulates this behaviour with the input parameter *Aggreg*. As the input you have to give the number of individuals placed together. The actual number of individuals placed is then either this number (*RndHet*: *fix*) or a normally distributed random number with mean *Aggreg* and variance *Aggreg* (*RndHet*: *rnd)*. The *Aggreg* value is typically similar to the value of the Lloyd index of mean crowding (see below) and realistic parameter values have therefore the same magnitude (0 to more than 10). ***NeutralCom* does not place all individuals of a species simultaneously during program run but places each individual or several individuals (as defined by *Aggreg* and *RndHet,* see below) sequentially.** For instance, of a species with in total 100 individuals and a value of *Aggreg* of 5 *NeutralCom* might place in the first step 6 individuals, in a second step 3, in a third step 5 and so on until all individuals are placed into cells of the grid. This allows a much more flexible treatment of the placement process.

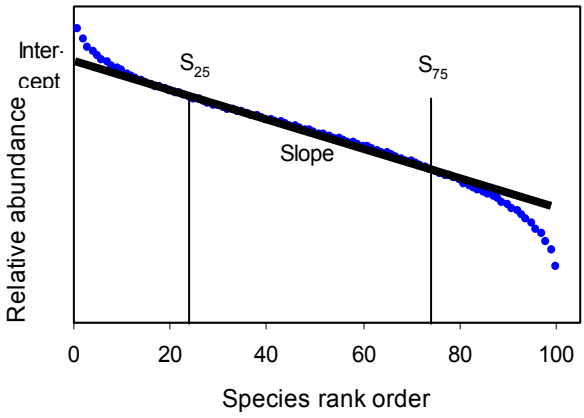

Figure 11: A relative abundance distribution (the blue data points stem from a Random fraction model of 100 species) can be characterized by a linear regression through the middle ranking 50% of species in a semi-logarithmic plot (see Ulrich 2001b, c). *NeutralCom* computes the slope and the intercept of this regression.

Table 4: The output file *Sample.txt* shows the total sample as defined by *maxSa*.

Characteristics of total sample

Areal	Samp-A	maxSa	minSa	Step-Sa	IndFac	Method	Species	maxD	minD	Dfluc	Flucmy	Flucvar	DivInd	Dfix	Model	Const=k	Const=X	TruncF
5	3	9	9	1	10	all	20	2	0.04	0	0	0	sha	0		0	0	0

DWD	SWD	Class	SWDmy	SW-Dvar	DWDmy	DWDvar	Aggr	Area-Het	FracM	Scale1	Scale2	Birth	Death	Spec-Rate	Rnd-Het	Grid-Het
linear	normal	30	15	2	0	0	0	0	rmd	2	1.5	0.1	0.1	0.2	rnd	fix

SpeNo	Weight	Ind	Density	StDev	log-2Dens	Lloyd	Density boundary	Genus	Family
1	2	14	1.5555	1.2570	0.637	1.01	50	2	3
2	4	24	2.6666	1.825742		1.094	50	3	1
3	6	13	1.4444	2.1659	0.531	2.556	50	1	2
4	8	13	1.4444	0.9558	0.531	0.746	50	5	3
5	12	13	1.4444	1.7069	0.531	1.704	50	6	4
6	16	19	2.1111	1.4487	1.078	0.997	50	1	1
7	24	21	2.3333	1.9436	1.222	1.265	50	1	2
8	32	15	1.6666	2.2110	0.737	2.16	50	3	3
9	42.67	12	1.3333	1.6329	0.415	1.75	50	3	1
10	53.33	13	1.4444	0.9558	0.531	0.746	50	5	2
11	64	16	1.7777	1.2272	0.83	0.914	50	7	1
12	96	23	2.5555	1.3425	1.354	0.885	50	2	3
13	128	11	1.2222	1.0304	0.29	0.893	50	3	2
14	192	22	2.4444	2.0608	1.29	1.302	50	2	1
15	256	9	1	1.4142	0	2	50	2	1
16	384	27	3	1.6996	1.585	0.988	50	1	1
17	512	21	2.3333	1.7638	1.222	1.143	50	6	1
18	1024	20	2.2222	1.6850	1.152	1.125	50	8	4
19	2048	4	0.4444	0.6849	-1.17	1.125	50	9	2
20	0	0	0	0	0	0	0	4	1
21	1.14	0	0	0	0	0	0	2	3
22	162-	0	0	0	0	0	0	6	1
23	383-	0	0	0	0	0	0	1	2
24	8.12	0	0	0	0	0	0	2	2
25	5245.7	0	0	0	0	0	0	2	2
26	1.04	1	0.1111	0.3142	-3.17	0	0	8	2
27	570-	0	0	0	0	0	0	9	1
28	145-	0	0	0	0	0	0	7	5
29	266-	1	0.1111	0.3142	-3.17	0	0	4	3
30	102.49	2	0.2222	0.4157	-2.17	0	0	3	2
31	229-	0	0	0	0	0	0	7	4
32	16.65	0	0	0	0	0	0	5	4
33	657-	2	0.2222	0.4157	-2.17	0	0	9	1
34	168-	1	0.1111	0.3142	-3.17	0	0	10	1
35	10.4	0	0	0	0	0	0	1	3
36	1.11	2	0.2222	0.4157	-2.17	0	0	2	2
37	636-	1	0.1111	0.3142	-3.17	0	0	5	1
38	107-	0	0	0	0	0	0	3	1
39	282-	0	0	0	0	0	0	8	1
40	465-	0	0	0	0	0	0	4	1
41	517-	0	0	0	0	0	0	5	3
42	8.91	0	0	0	0	0	0	11	4

No.	Ind.	Total bio-	SD lb	RAD	RAD	Diver-	Even-	Alpha	Theta	Area
26	320	145306.6	1.6552	-0.1213	-1.7394	2.9868	0.9167	6.69	380	9

3.4.3. Defining the grid and cell properties

Real habitats or local resource patches always differ more or less. For a realistic model differences in cell features have therefore to be included. *NeutralCom* offers for this task six options.

AreaHet defines the degree of heterogeneity of the grid. This means in this case the degree by which the cells of the grid differ. *NeutralCom* models this difference by the number of individuals N of a certain species that can be placed into one cell at each placing step. The actual number of individuals placed is then either this number (*RndHet*: *fix*) or a normally distributed random number with mean *AreaHet* and variance *AreaHet* (*RndHet*: *rnd)*.

You can choose whether the grids structure is identical for all species (*Gridhet*: *fix*) or whether the grid structure is computed anew for each species (*GridHet*: *rnd*). The latter option results in a certain randomization of total species spatial distribution patterns.

Fig. 7 shows a simple example from a grid of 20 * 20 cells and *AreaHet* = 10. The input value *AreaHet* = 0 gives all cells the same properties and one individual will be placed during each step if not otherwise defined by *Aggreg* and *Range*.

PatchOccupancy is able to compute a wide range of different habitat types including fractal landscapes. The latter can be computed in two different ways, by a superposition of trigonometric functions (*FracM*: *super*) or by random midpoint displacement (*FracM*: *rmd*).

In the first case the basic computing procedure is a superposition process (Burrough 1983) that is similar to a discrete Fourier transformation (Hastings and Sugihara 1993). In this case this process is based on a superposition of trigonometric functions of the general form

$$N = \sum_{i=1}^{k(scale1)} \frac{\text{AreaHet}}{\text{scale1}^{k-1}} \left(\cos(\frac{x}{\text{scale2}^k}) + \sin(\frac{y}{\text{scale2}^k}) \right)$$

In this equation x and y denote the coordinates of the grid, *Areahet*, *scale1*, and *scale2*

Table 5: *Analysis.txt* is an output file that gives basic features of the sampling process and computes additional statistics.

First row: Area samples with 1,2,...n sample units
Second row: No. species in exactly 1, 2...n sample units
Third row: Mean number of new species per sample unit
Fourth row: Number of species found only in that sample (endemics)
Fifth row: Cumulative number of new species per sample unit
Sixth row: No. species of assemblage, No. species/unit area, No. species sampled,
rel. species density (species per unit area /No. species in the assemblage),
Bootstrap summand, total number of samples, Species found in exactly one and in two sample units
Seventh row: SARS and EARS: SAR intercept, slope and variance explanation of the power function, and the exponential model, and the power function EAR
Eight row: Variances of cell properties (ninth) and surface length (tenth) with increment (ruler length) of factor 2 (of the two diagonals of the grid)
Ninth, tenth and eleventh row : Number of binary scaling classes, number of classes included in the regression, intercept of log(variance) - log(increment) plot, slope, fractal dimension, intercept of log(surface length) - log(ruler length) plot, slope, fractal dimension

Areal	SampA	maxSa	minSa	StepSa	IndFac	Method	Species	maxD	minD	Dfluc	Flucmy	Flucvar	DivInd	Dfix	Model	Const=	Const=	TruncF
5	3	9	9	1	10	all	20	2	0.04	0	0	0	sha	0		0	0	0
DWD	SWD	Class	SWDmy	SWDvar	DWDmy	DWDvar	Aggr	Area-Het	FracM	Scale1	Scale2	Birth	Death	Spec-Rate	RndHet	GridHet		
linear	normal	30	15	2	0	0	0	0	rmd	2	1.5	0.1	0.1	0.2	rnd	fix		
1	2	3	4	5	6	7	8	9	10	11	12							
4	3	1	3	2	1	9	3	0										
13.9	5.1	2.1	1.1	1	0.6	0.5	0.6	1										
0	0	0	0	0	0	0	0	0										
13.9	19	21.1	22.2	23.2	23.9	24.4	25	26										
19	13.9	26	0.7289	1.7	9	4	3											
14.944	0.26273	0.9496	14.72063	5.15273	0.98085	0	0	0										
1	1																	
0.0094	0.01642																	
0.4111	1.34337																	
3	2	0	0	2	0	0	1											

are the parameters (see below), N is the number of individuals placed, and *k(scale)* the number of iterations that depends on the grid size and the scaling factor *scale1 (k = 2ln(Area)/ln(scale1)*.

In the case of the random midpoint displacement algorithm area sizes n are adjusted to values of 2^n+1. Because this algorithm results in positive and negative values for landscape roughness values are shifted by *value*=*value**3**scale1* + *scale1*. As in the previous method input parameters are *Areahet*, *scale1*, and *scale2*. *Scale1* defines the fractal dimension ($2 \leq$ scale1 ≤ 3) and is then transormed to the Hurst exponent by H = 3—D. *Scale2* defines the variances of the random offset during the displacement procedure.

Figure 9 shows the diagonal through a grid of 1,000,000 cells (*Area* = 1000) under three different magnifications generated by the superposition method. From the two diagonals of the grid *NeutralCom* computes the fractal dimension of the grid from a semivariogram (Hurst 1951, Burrough 1983, Hastings and Sugihara 1993) that uses the relation between variance per distance h and distance

$$Variance(h) \propto h^{2H}$$

H is the so-called Hurst exponent and the fractal dimension D is related to H by D = 2 - H.

Additionally, D is computed from the ruler length method based on the relation

$$Length(h) \propto h^{1-D}$$

AreaHet defines how many individuals are in the mean placed at each placing step and has to be a non-negative real number. An input of 0 results in a homogeneous grid.

Scale1 and *Scale2* are two positive parameters that define the structural properties of the grid. Setting *Scale1* to 1 defines a non-fractal grid.

RndHet defines whether at each step exactly N individuals are placed (input: *fix*) or whether this number is a normally distributed random variable with mean N and variance N.

DivInd defines the diversity and evenness statistics *NeutralCom* uses. You may choose three diversity indices with it's associated evenness measures:

a) Shannon – Wiener (*sha*):

$$H_{sha} = -\sum p_i \ln(p_i); E_{sha} = H_{sha} / \ln(S)$$

b) Simpson (*simp*):

$$H_{simp} = 1/\sum p_i^2; E_{simp} = H_{simp} / S$$

c) Smith – Wilson (*evar*) (Smith and Wilson 1996):

$$E_{\text{var}} = 1 - \frac{2}{\pi}\arctan\left(\frac{\sum_{i=1}^{S}(\ln(p_i) - \frac{\sum_{i=1}^{S}\ln(p_i)}{S})^2}{S}\right);$$

$$H_{\text{var}} = E_{\text{var}} \times S$$

E_{var} is equivalent to the arctan transformed Gaussian width. For these three indices the program gives also standard deviations obtained from the respective values of each replicate. Additionally, *NeutralCom* computes the α-diversity index (Magurran 1988, Rosenzweig 1995).

If you compute species—area relations from a summation process of independent cells (the *Method dep* option; see Fig. 6) the outcome depends highly on the ordering of cells. To minimize this effect *NeutralCom* reshuffles the sampling order 20 times. Colwell and Coddington (1994) showed that 20 such reshufflings are sufficient to get stable SAR patterns.

3.4.3. Taxonomic level

NeutralCom offers a simple (and simplified) tool to study higher taxonomic levels. At the beginning each species is at random assigned to two taxonomic levels termed genus and family. During program run new species get then the taxonomic status of their progenitors. However, with the same probability as for the speciation events new taxa are generated.

4. Using real communities for sampling

NeutralCom is able to read extern files in the text-format that contain data about abundances and body weights (optional) of real communities. It also reads the *Assemblage.txt* file generated by the program RAD (Ulrich 2002b) that generates theoretical relative abundance distributions. *NeutralCom* asks at the beginning what type of file to be used. In the case of files other than *Assemblage.txt* the file has to contain a * as the first sign before the beginning of the data. Because *NeutralCom* can run via a batch file an unlimited number of real community data can be processed during one program run. **The data have to be sorted according to density!**

Table 2 shows an example of such a batch file. Note that body weights are later printed in the fixed format Fn.2. It is therefore recommended to use an appropriate unit of measurement for body weights.

In Table 2 densities are given as numbers of individuals. However, *NeutralCom* does not know how density data are given. But, you can influence the way of data processing. If the first species in the list has a density equal to or above 1 *NeutralCom* assumes that the data are given as individuals. It then adjusts this density to the maximum defined by *maxD*. If the first species has a value below 1 *NeutralCom* assumes that densities are given as relative densities. In this case the program first adjusts the data in such a way that the least abundant species has exactly 1 individual. In a next step these modified abundances are again adjusted according to the settings in *maxD*. In the example of community A of Tab. 2 the first species has a density value D > 1. Given that you defined *maxD* as 10 individuals per cell and use a grid of 100 cells (*Area* = 10) total individual numbers per species S_i to be placed in the grid are computed from $S_i = D_i*(100*10/D_1)$. The first species gets therefore the individual number 1000, the second 590. If you want to use the individual numbers of the input file you have therefore to set *maxD* to $D_1/(Area*Area)$.

In community B of Tab. 2 D_1 is less than 1. In this case all data are first divided through the density of the least abundant species D_S. Total individual numbers per species are therefore again computed from $S_i = D_i*(100*10/D_1)$. In our example the first species gets again 1000 and the second species 623 individuals. The least abundant species gets 96 individuals. If you want the least abundant species to have exactly 1 individual you would have to set *maxD* to 1/96 = 0.0104.

5. The output files

5.1. Distribution.txt

NeutralCom generates 6 output files: *Distribution.txt*, *Sample.txt*, *Cells.txt*, *Analysis.txt*, *Succession.txt*, and *Grid.txt*.

Distribution.txt (Tab. 3) gives the original community used or generated by the program. Table 3 shows an example of this file.

The first five lines in this file show once again the input parameters. They contain four variables *Model*, *Const=K*, *Const=X*, and *TruncF* that refer to the case when the program uses a theoretical relative abundance distribution generated by the program RAD (Ulrich 2002b).

The next line contains the number of species per $\log_2$ weight class. In this example 30 classes were defined, but for 20 species distributed according to a normal species—body weight distribution with mean = 15 and a standard deviation of 1 only 11 weight classes are possible.

Then, *NeutralCom* gives the assigned body weights per species (*Weight*), the total number of individuals per species before placing them into the grid (*OrigInd*), the total number of individuals per species after placing (*AssInd*), and the theoretical

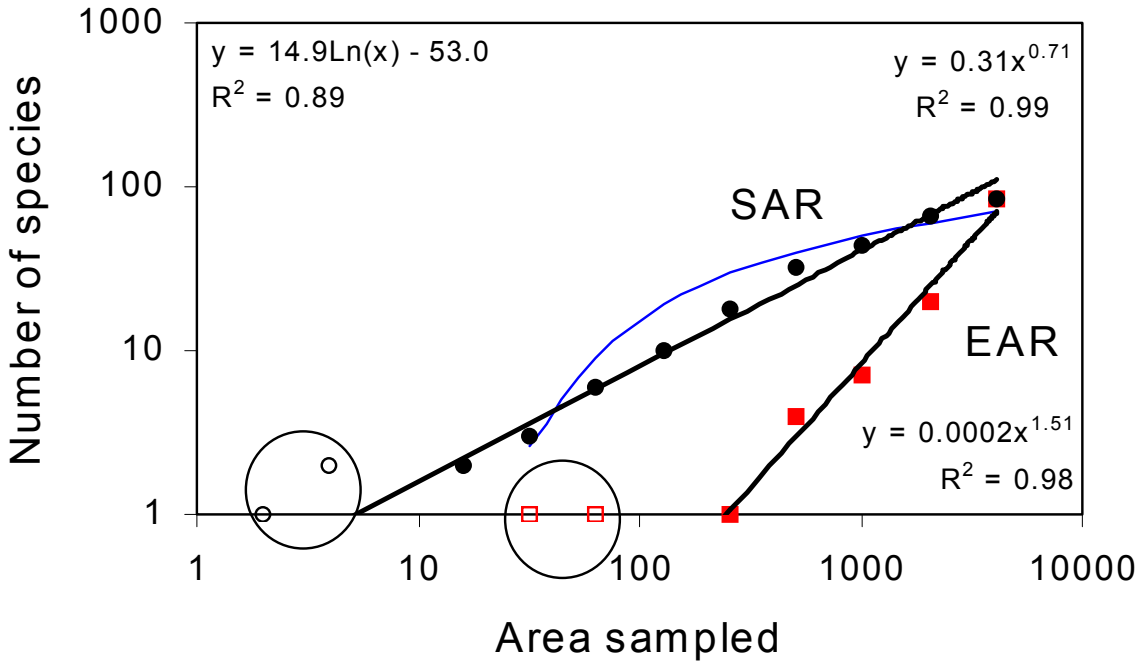

Fig. 12: An example how *NeutralCom* computes species - area and endemics - area relations. Logarithmic and power function models are fitted to the data set. The four data points marked with circles are not included in the computation.

upper abundance limit per species as defined by the density—body weight distribution (*Density boundary*). *OrigInd* is nearly always lower then the density boundary except in cases where the DWD model or the model used for density fluctuations had been a normal or a log-normal distribution.

Note that *OrigInd* and *AssInd* sometimes differ. That means not all individuals could be placed into the grid due to the limitations set by the grid property parameters.

Each species got originally a species number (*SpeNo*) with which it can be identified. *NeutralCom* does not sort the species list after placing and sampling.

The last output lines contain some statistics about the assemblage after placing. The first value is the number of species placed into the grid. This value is often smaller than the original species number of the community. Then comes the total number of individuals of the assemblage after placing. The product of individual number and species body weight gives the biomass. *NeutralCom* sums up these biomasses and prints the sum (*Total biomass*). In cases where no species biomasses are available a zero value is printed.

The next output value is the Gaussian weight of the assemblage after placing (SD lb(dens). The Gaussian weight is the standard deviation of $\log_2$ transformed densities of each species and a measure of community structure (see Sugihara 1980 and Tokeshi 1993). *RADslope* and *RADinter* give the slope and the intercept of the relative abundance distribution of the assemblage after placing (Ulrich 2001b, c, 2002a)(Fig. 11).

The next output values are a diversity and the accompanying evenness index (see above). At last the α-diversity index and Hubbell's fundamental biodiversity number θ are printed. α is computed from

$$S = \alpha \ln(1 + \frac{N}{\alpha})$$

with S being the total number of species and N the

Table 6: The optional output file *Succession.txt* gives the whole sample matrix, the coordinates of the sampled cells and the numbers of individuals per sampled cell and per species at each stage of a successional process.

Characteristics of whole community																	
Areal	SampA	maxSa	minSa	StepSa	IndFac	Method	Species	maxD	minD	Dfluc	Flucmy	Flucvar	DivInd	Dfix	Model	Const=	Const=	TruncF
5	3	9	9	1	10	all	20	2	0.04	0	0	0	sha	0		0	0	0
DWD	SWD	Class	SWDm	SW-	DWDm	DWDva	Aggr	Area-	FracM	Scale1	Scale2	Birth	Death	Spec-	Rnd-	GridHet		
linear	normal	30	15	2	0	0	0	0	rmd	2	1.5	0.1	0.1	0.2	rnd	fix		
Coordinates of Cells																		
	1	2	3	1	2	3	1	2	3									
	1	1	1	2	2	2	3	3	3									
Species Data, each row is the sampling of one of the 20 species																		
1	2	1	0	1	0	1	9	1	4									
2	2	0	1	0	0	1	3	3	2									
3	1	1	2	2	0	0	3	2	1									
4	2	3	2	2	0	3	1	1	3									
5	0	2	2	3	0	3	3	1	1									
6	0	1	4	3	0	4	3	1	2									
7	5	1	0	2	0	1	1	2	1									
8	2	2	5	2	0	1	5	3	0									
9	1	2	1	3	0	3	2	1	2									
10	1	1	1	2	0	1	2	2	3									
11	3	4	1	2	0	1	2	2	2									
12	3	2	0	1	0	3	3	3	2									
13	1	3	2	1	0	1	4	0	2									
14	0	1	1	3	0	2	0	1	1									
15	0	2	0	2	0	4	1	3	2									
16	5	2	1	3	0	4	2	1	2									
17	1	0	1	1	0	1	3	3	3									
18	3	2	1	2	0	1	1	2	0									
19	4	1	0	2	0	3	1	0	5									
20	0	0	0	0	0	0	0	0	0									
S	15	17	14	18	0	18	18	17	17									
N	36	31	25	37	0	38	49	32	38									
Alpha	9.66	15.44	13.14	13.83	0	13.38	10.27	14.72	11.82									
Cycle:	1 Individuals handled:		437															
1	0	1	0	1	0	0	7	1	4									
2	2	0	1	0	0	1	2	2	4									
3	2	0	2	2	0	0	4	1	1									
4	2	3	3	2	0	5	0	1	2									
5	1	2	3	2	1	2	6	2	1									
6	0	2	3	3	0	4	2	1	2									
7	6	1	0	1	0	0	1	2	1									
8	1	3	2	1	0	1	3	3	0									
9	1	2	1	3	0	4	2	1	3									
10	1	2	1	2	0	1	2	2	4									
11	2	2	1	4	0	1	3	1	0									
12	3	2	0	1	0	4	3	4	2									
13	1	3	3	1	0	1	4	0	2									
14	0	3	2	3	0	2	0	1	1									
15	0	2	0	1	0	6	1	3	2									
16	5	0	1	4	0	4	2	1	2									
17	1	0	2	1	0	0	4	3	3									
18	4	2	1	4	0	0	1	3	0									
19	4	1	0	1	0	2	2	0	4									
20	0	0	0	0	0	0	0	0	0									
21	0	0	0	0	0	0	0	0	0									
S	15	15	14	18	1	14	17	17	16									
N	36	31	25	37	1	38	49	32	38									
Alpha	9.66	11.45	13.14	13.83	1000	8.01	9.23	14.72	10.42									

Table 7: The optional output file *Cells.txt* gives the whole sample matrix, the coordinates of the sampled cells and basic community statistics.

Areal	SampA	maxSa	minSa	StepSa	IndFac	Method	Species	maxD	minD	Dfluc	Flucmy	Flucvar	DivInd	Dfix	Model	Const=k	Const=X	TruncF
5	3	9	9	1	10	all	20	2	0.04	0	0	0	sha	0		0	0	0
DWD	SWD	Class	SWDmy	SW-Dvar	DWDmy	DWDvar	Aggr	Area-Het	FracM	Scale1	Scale2	Birth	Death	Spec-Rate	Rnd-Het	GridHet		
linear	normal	30	15	2	0	0	0	0	rmd	2	1.5	0.1	0.1	0.2	rnd	fix		

Coordinates of Cells

	1	2	3	1	2	3	1	2	3
	1	1	1	2	2	2	3	3	3

Births: 2253. Deaths: 2254. Immigr.: 854. Emigr.: 875.

Species Data, each row is the sampling of one of the 42 species

1	1	2	0	4	1	1	2	0	3
2	4	4	3	1	5	5	1	1	0
3	2	2	0	0	0	7	2	0	0
4	0	2	1	2	3	0	1	2	2
5	0	0	0	0	2	0	4	4	3
6	0	3	2	4	0	4	2	3	1
7	2	2	0	0	2	1	5	6	3
8	1	0	0	1	3	0	7	0	3
9	3	5	1	1	0	0	0	0	2
10	1	0	2	3	1	2	2	2	0
11	4	0	2	2	2	2	3	0	1
12	4	1	3	3	2	4	2	0	4
13	2	0	0	0	3	1	1	2	2
14	1	6	1	5	2	0	4	0	3
15	3	0	0	0	1	1	4	0	0
16	5	0	4	5	2	2	4	4	1
17	0	3	3	0	1	3	2	6	3
18	0	0	3	5	3	4	1	1	3
19	1	0	0	0	0	0	1	0	2
20	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0
26	0	0	0	0	1	0	0	0	0
27	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	1
30	0	0	0	0	0	1	0	0	1
31	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0
33	1	0	0	1	0	0	0	0	0
34	0	0	0	0	0	0	0	1	0
35	0	0	0	0	0	0	0	0	0
36	1	1	0	0	0	0	0	0	0
37	0	0	0	0	0	0	1	0	0
38	0	0	0	0	0	0	0	0	0
39	0	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0
41	0	0	0	0	0	0	0	0	0
42	0	0	0	0	0	0	0	0	0
S	16	11	11	13	16	14	19	11	17
N	36	31	25	37	34	38	49	32	38
Alpha	11.04	6.09	7.51	7.14	11.8	8.01	11.4	5.93	11.82

total individual number. Note that α should asymptotically be identical with θ.

5.2. Sample.txt

In the case of a sample 3 other files are printed. *Sample.txt* gives the basic features of the sampled community. An example contains Tab. 4.

Again the first four output lines contain the program settings. Then, the program prints the species numbers (the same as in Table 3) and the body weights assigned by the program. *Ind* contains the total number of individuals found in all cells as defined by *maxSa*. *Density* gives the mean density per sample unit and *StDev* its standard deviation. The latter is only given in cases when *Method* is defined as *dep*. *Log2dens* gives the logarithm ($\log_2$) of *Density*. Lloyd computes the Lloyd index of aggregation. This index is defined by

$$L = \frac{StDev^2}{Density^2} - \frac{1}{Density} + 1$$

At last, the program gives the density boundary. This is the maximum number of individuals per sampling unit.

At the end of the output *NeutralCom* gives the number of species and the number of individuals sampled, the total biomass of the sample [=Σ (Density * Weight)], the Gaussian weight, and the area sampled. *Diversity*, *Evenness*, α, and θ refer to the diversity statistics chosen. RAD slope and RAD intercept again refer to the regression line through the middle ranking 50% of species as explained in Figure 11.

Table 8: The output file *Grid.txt* prints the maximum number of individuals to be placed at each placing step for both diagonals (1. Diag from 1,1 to *area, area*; 2. Diag from 1,*area* to *area*,1)

Maximum number of individuals that can be placed into the cells of both grid diagonals at each placing step

Cell	1. Diag	2. Diag
1	4	4
2	3	5
3	2	7
4	3	6
5	2	4
6	5	5
7	7	2
8	4	1
9	1	3
10	3	4

5.3. Analysis.txt

The third basic output file is *Analysis.txt* (Tab. 5). *Analysis.txt* prints basic features and statistics of the sampling process. Again, at the beginning *NeutralCom* prints the program settings. Then the program prints the area sampled with n sample units. In the case of Tab. 5 the maximum sample size was defined as 12 and the sample unit is 1, that means a single cell. The maximum area is therefore 12 cells, the last value in the respective line.

The next lines print species numbers and some statistics of the sampling process. These are

1. The number of species found in exactly 1, 2, … n sampling units
2. The mean number of new species per sampling unit,
3. The number of species found only in that sample (the endemic species) (this option refers only to the sampling methods *oct* and *ind*), and
4. The total number of species found in the area given above. This latter line together with the area line, of course, define the species—area relationship.

The next line gives 8 basic values:

1. The number of species of the assemblage (assemblage refers always to the species actually placed into the grid),
2. The mean number of species found per sampling unit at this step,
3. The number of species found at this step,
4. The relative species density. This is the mean number of species found per unit of area divided through the total number of species of the assemblage,
5. The next value is the so called bootstrap summand, necessary to estimate species number via a bootstrap method. This value is defined as:

$$bootstrap = S_S + \sum_{i=1}^{S_S} (1 - p_i)^n$$

(Smith and van Belle 1984) with S_S being the number of species found in n samples and p_i its relative density.

6. The total number of samples per step

7. The number of species found in exactly 1 and 2 samples.

The next lines give the power function and the logarithmic type of species area—relationships as derived from each sample step. The first three values are the intercept, the slope, and the variance explanation of the power function model, the next three values are the intercept, the factor, and again the variance explanation of the logarithmic SAR. Additionally, *NeutralCom* computes a so called endemics—area relation (EAR) (Fig. 12). The last three values of this line are the intercept, the slope and the variance explanation of a power function EAR model. The input data are the values printed in the endemics line (Nr. 3 above). All values refer to least square fits. Starting point for the computations is always the highest non-zero sample (Fig. 12).

The above values are given for each sample step. By this way *NeutralCom* allows an easy estimate of the sample effectiveness and how good for instance species numbers, diversities or SAR patterns are approximated by different sample sizes.

At last, the program prints the distance h (measured as cells) of both grid diagonals , the associated variance in cell properties (to define a semivariogram), and the total surface length. The next line contains first the number of binary ruler length classes h and the number of ruler length classes used to computes the fractal dimension D. Because *NeutralCom* uses a superposition of trigonometric functions one grid might contain more than one scaling regions. Therefore, the program first looks for steps in the variance distribution and adjusts then the number of ruler length classes.

The next values of this output line are, the intercept of the ln (variance) - ln (h) regression, the associated slope and the fractal dimension. Then follow the intercept and the slope of the ln (surface length) - ln (h) regression and again the fractal dimension. In the case of Tab. 5 the grid is much too small to estimated fractal dimensions. In this case the program returns values of 1 and 2. In general, you will need grids of more than 1,000,000 cells for a sufficiently precise estimate of grid surface structure. Unfortunately, such values result in extraordinary high computing times. Note that sometimes computed fractal dimensions are outside the possible range. This happens when grid structures are chosen with either symmetrical features or several scaling regions. It is therefore recommended to check grid structure using the output file *Grid.txt* that gives the cell properties of both diagonals.

5.4. Succession.txt

Succession.txt is an optional file and gives the numbers if individuals per species found in each sample (Tab. 6) after the placing process and at each stage of a successional process. The output shows at most 100 such successional processes regularly spaced over the whole process as defined by density and the number of cycles.

Again, *Succession.txt* contains the program settings. If *Method* was defined as ‘dep’ it then gives the coordinates of the cells sampled. In the cases of *Method* = ‘seq’ and *Method* = ‘ind’ *Succession.txt* gives in this place the area sampled. The next lines contain the number of individuals per species and cell at the beginning of the process and at each successional stage. Additionally for each stage three basic measures of community structure are given, the number of species S, the number of individuals N, and the α-diversity index. Note that α is ∞ if S = N. In this case α is set to 1000.

5.5 Cells.txt

An additional optional file is *Cells.txt*. This file is identical to *Succession.txt* but shows the individual numbers after the model run (Tab. 7). *Cells.txt* and *Succession.txt* together allow therefore comparisons to be made about the effect of birth, death, immigration, emigration, and speciation on community structure.

5.6 Grid.txt

The last output file is *Grid.txt*. It gives the maximum number of individuals that can be placed (at each placing step) into the cells of both diagonals of the whole grid (Tab. 8).

6. Some problems with this version

The definition of large grids and larger species numbers may result in extraordinary large computing times. Define therefore these values not larger than necessary to solve a given problem. Care has to be taken that the initial program setting for the SWD and the DWD are realistic, Unrealistic settings result later in assemblage structures that have nothing in common with structures of real world communities. In a few cases, especially at low species or individual numbers, unrealistic initial settings force the program into runtime errors or endless loops.

7. System requirements

NeutralCom is written in FORTRAN 95 and runs under Windows 9.x, and XP. Computation abilities are only limited by the computer's memory.

8. Citing *NeutralCom*

NeutralCom is freeware but nevertheless if you use *NeutralCom* in scientific work you should cite *NeutralCom* as follows:

Ulrich W. 2002 - NeutralCom – a FORTRAN program for generating neutral macroecological models - www.uni.torun.pl/~ulrichw

9. Acknowledgements

The development of this program was supported by a grant of the Polish Science Committee (KBN, 3 F04F 03422).

10. References

Bell G. 2000 - The distribution of abundance in neutral communities - Am. Nat. 155: 606-617.

Bell G. 2001 - Neutral macroecology - Science 293: 2413-2418.

Burrough P. A. 1983 - Multiscale sources of spatial variation in soil II. A non-Brownian fractal model and its application in soil survey - J. Soil Sci. 34: 599-620.

Caswell H. 1976 - Community structure: a neutral model analysis - Ecol. Monogr. 46: 327-354.

Colwell R. K., Coddington J. A. 1994 - Estimating terrestrial biodiversity through extrapolation - Phil. Trans. R. Soc. Lond. B: 345: 101-118.

Condit R. et al.. 2000 - Spatial patterns in the distribution of tropical tree species - Science 288: 1414-1418.

Condit R. et al. 2002 - Beta-diversity in tropical forest trees - Science 295: 666-669.

Currie D. J. 1993 - What shape is the relationship between body mass and population density - Oikos 66: 353.

Currie D. J., Fritz J. T. 1993 - Global patterns of animal abundance and species energy use - Oikos 67: 56-68.

Cyr H., Peters R. H., Downing J. A. 1997 - Population density and community size structure: comparison of aquatic and terrestrial systems - Oikos 80: 139-149.

Gaston K. J. 1993 - Comparing animals and automobiles: a vehicle for understanding body size and abundance relationships in species assemblages? - Oikos 66: 172-179.

Gaston K. 2000 - Global patterns in biodiversity - Nature 405: 220-227.

Gotelli N. J., Graves G. R. 1996 - Null Models in Ecology - Smith. Inst. Press, Washington D.C.

Hanski I. 1991 - Single-species metapopulation dynamics: concepts, models and observations - Biol. J. Linnean Soc. 42: 17-38.

Hanski I. 1999 - Metapopulation ecology - Oxford (Univ. Press).

Hanski I., Kouki J., Halkka A. 1993 - Three explanations of the positive relationship between distribution and abundance of species (In: Species Diversity in

Ecological Communities, Eds.: R. E. Ricklefs, D. Schluter) - Univ. Press, Chicago, pp. 108-116.

Hanski I., Gilpin M. E. (Eds.) 1997 - Metapopulation Biology - London Academic Press, 512 pp.

Hastings H. M., Sugihara G. 1993 - Fractals - A User's Guide for the Natural Sciences - Oxford Univ. Press.

Hubbell S. P. 1997 - A unified theory of biogeography and relative species abundance and ist application to tropical rain forest and coral reefs - Coral Reefs 16, Suppl.: S9-S21.

Hubbell S. P. 2001 - The unified theory of biogeography and biodiversity - Princeton (Univ. Press).

Hurst H. E. 1951 - Long-term storage capacity of reservoirs - Trans. Am. Soc. Civil Eng. 116: 770-808.

Kimura M. 1985 - The Neutral Theory of Molecular Evolution - Cambridge Univ. Press.

Kimura M., Naoyuki T., Crow J. F 1994 - Population Genetics, Molecular Evolution, and the Neutral Theory - Chicago Univ. press

Leitner W. A., Rosenzweig M. L. 1997 - Nested species-area curves and stochastic sampling: a new theory - Oikos 79: 503-512.

Magurran A. E. 1988 - Ecological diversity and its measurement - Princeton (Univ. Press), pp. 1-179.

May R. M. 1975 - Patterns of species abundance and diversity (in: Ecology and evolution of communities, Eds.: M. L. Cody, J. M. Diamond) - Belknap (Cambridge), pp. 81-120.

Novotny V., Kindlmann P. 1996 - Distribution of body sizes in arthropod taxa and communities - Oikos 75: 75-82.

Smith E. P., van Belle G. 1984 - Nonparametric estimation of species richness - Biometrics 40: 119-129.

Smith B., Wilson, J. B. 1996 - A consumer's guide to evenness indices - Oikos 76: 70-82.

Sugihara G. 1980 - Minimal community structure: an explanation of species abundance patterns - Am. Nat. 116: 770-787.

Tokeshi M. 1993 – Species abundance patterns and community structure – Adv. Ecol. Res. 24: 111-186.

Tokeshi M. 1996 - Power fraction: a new explanation of relative abundance patterns in species-rich assemblages - Oikos 75: 543-550.

Wilson W. B., Gitay H., Steel J. B., King W. McG. 1998 - Relative abundance distributions in plant communities: effects of species richness and spatial scale - J. Veg. Sci. 9: 213-220.

Ulrich W. 1999a - Estimating species numbers by extrapolation I: Comparing the performance of various estimators using large model assemblages - Pol. J. Ecol. 47: 271-291.

Ulrich W. 1999b - Estimating species numbers by extrapolation II: determining the minimum sample size to obtain a certain fraction of species in a community - Pol. J. Ecol. 47: 293-305.

Ulrich W. 1999c - Abundance, biomass and density boundaries in the Hymenoptera: analysis of the abundance - body size relationship and differences between forest and open landscape habitats - Polish J. Ecol 47: 73-86.

Ulrich W. 2000a - On species - area relationships I: fit of power function and exponential model - Pol. J. Ecol. 48: 3-20.

Ulrich W. 2000b - On species - area relationships II: slope and factor values of power function and exponential model - Pol. J. Ecol. 48: 21-35.

Ulrich W. 2000c - On species - area relationships III: the intercept of the power function and the exponential model - Pol. J. Ecol. 48: 37-48.

Ulrich W. 2000d - Species area relations generated by theoretical relative abundance distributions: parameter values, model fit and relation to species saturation studies - Pol. J. Ecol. 48: 67-78.

Ulrich W. 2001a - Relative abundance distributions of species: The need to have a new look at them - Pol. J. Ecol.: 393-407.

Ulrich W. 2001b - Models of relative abundance distributions I: model fitting by stochastic models - Pol. J. Ecol. 49: 145-157.

Ulrich W. 2001c - Models of relative abundance distributions II: diversity and evenness -statistics - Pol. J. Ecol. 49: 159-175

Ulrich W. 2002a – PatchOccupancy – a FORTRAN program for the study of species spatial distributions – www.uni.torun.pl/~ulrichw.

Ulrich W. 2002b - RAD – a FORTRAN program for the study of relative abundance distributions – www.uni.torun.pl/~ulrichw.

Ulrich W. 2003—Allometric biomass - species diversity relationships in animals: an outcome of basic underlying ecological distributions? - unpubl. Manuscript.

Whitfield J. 2002 - Ecology: Neutrality versus the niche - Nature 417: 480-482.

Wright D. H. 1991 - Correlations between incidence and abundance are expected by chance - J. Biogeogr. 18: 463-466.

Ziv Y. 2000 - On the scaling of habitat specificity with body size - Ecology 81: 2932-2938.