Appendix A: Simulation details

The multi-species Droop model (given by eqns 3, 4, 5) contains a large number of parameters and, hence, has many degrees of freedom: the maximum growth rates r_i and the mortality rates m_i of the n species; the minimum subsistence quotas q_{ji}; the maximum uptake rates v_{ji} and the half saturation constants for uptake K_{ji}; the flow rate D and the resource supply rates S_j. It can be shown analytically (Revilla and Weissing, submitted) that the dynamics of the Droop model depends crucially on the ranking of the species with respect to resource requirements R^*_{ji} and consumption patterns $\hat{f}_{ji}$ at equilibrium (see Table 1). For this reason we wanted to consider different scenarios regarding resource requirements and consumption. To this end, we started by imposing a certain sign pattern of the R^*_{ji} values and on the matrix $\hat{f}_{ji} = f_{ji}(\hat{R}_j)$ as in Table 1. To achieve such patterns the model parameters were chosen by a "reverse engineering" approach along the lines indicated below:

1. We set the flow rate D, maximum growth rates r_i and the mortality rates m_i. In most simulations, all r_i were equal: $r_i = r$ and all mortality rates were set equal to D (see the text for a motivation).

2. Resource requirements were randomly chosen from an interval (R^*_{low}, R^*_{high}) and arranged to form a matrix R^*_{ji} with the structure depicted in Table 1. Subsequently, the resource levels at equilibrium are set by $\hat{R}_j = R^*_{jj}$.

3. Consumption rates at equilibrium were randomly chosen from an interval $(\hat{f}_{low}, \hat{f}_{high})$ and arranged to give a $\hat{f}_{ji}$ matrix with a given hierarchy as in Table 1.

4. The half saturation constants K_{ji} were chosen at random, with the sole restriction that $K_{ji} > 0$.

5. The maximum uptake rates v_{ji} were determined from eqn (5): $v_{ji} = (1 + K_{ji}/\hat{R}_j)\hat{f}_{ji}$ where $\hat{R}_j = R^*_{jj}$. Notice that this results in a positive correlation between the parameters v_{ji} and K_{ji}, leading to resource uptake functions $f_{ji}(R_j)$ that tend to intersect as in Fig. 7.

6. Internal resource requirements are obtained from (7) as $Q^*_{ji} = v_{ji}R^*_{ji}/m_i(K_{ji} + R^*_{ji})$.

7. The minimum subsistence quotas are now obtained from (6) as $q_{ji} = (1 - m_i/r_i)Q^*_{ji}$.

8. Positive equilibrium densities $\hat{N}_i$ were chosen at random.

9. The supply rates S_j were obtained by setting the right-hand side of eqn (3c) equal to zero: $S_j = \hat{R}_j + (1/D)\sum \hat{f}_{ji}\hat{N}_j$.

The alternative approach of starting by choosing all parameters at random has two disadvantages. First, many combinations would result in a system without community equilibrium or a system where the community equilibrium is not attainable. In our approach steps 2, 8 and 9 guarantee that a community equilibrium exists and that the supply point fall inside the wedge in Fig. 1B, implying that the community equilibrium cab be reached. Second, and more importantly, the huge number $((k!)^{2k})$ of qualitatively different configurations of resource requirements and consumption patterns implies that any new set of parameter combinations would lead to a system falling into a new category. Accordingly an astronomic number of simulations would be required in order to draw any general conclusion on the system dynamics.

The system was initialized at the community equilibrium plus a random perturbation on the resources, i.e. $N_i(0) = \hat{N}_i$, $Q_{ji}(0) = \hat{Q}_{ji}$, $R_j(0) = \hat{R}_j + \varepsilon_j$. For each scenario, we replicated the above procedure 10000 to 50000 times, and we performed the corresponding numerical integration for $t = 10000$ to 50000 days.

Numerical integration was performed in the C programming language using the 4th order Runge-Kutta solver from the GNU Scientific Library (GSL) `http://www.gnu.org/software/gsl/`. Each time unit of the

simulation (day) is fractioned into 1000 time steps by default unless adaptive step size control takes place with absolute error tolerance of 10^{-6}. The minimum operator of (4) is evaluated for each single time step. We checked the robustness of the numerical procedure by rerunning a large number of simulations with different time steps and error tolerances, and by replacing the minimum operator in (2) and (4) by a differentiable function.

Appendix B: Parameter values

Parameter values are listed in the following order:

1. Flow rate (D), maximum growth rate (r) and resource supplies (S_j).
2. The parameters K_{ji}, v_{ji} and q_{ji} are given by three matrices **K**, **V** and **Q**, where the rows indicate resources and columns indicate species.
3. Initial conditions for the species densities. If the species is a late invader, the time of invasion is placed between parentheses.

Figure 2:
$D = 0.50, r = 1, S_1 = S_2 = S_3 = 10$

	A			B			C			D		
K	0.47	0.66	0.73	0.43	0.21	0.24	0.43	0.21	0.24	0.47	0.75	0.84
	0.41	0.78	0.21	0.29	0.52	0.25	0.29	0.52	0.25	0.10	0.56	0.51
	0.16	0.82	0.16	0.41	0.31	0.46	0.41	0.31	0.46	0.54	0.43	0.21
V	2.17	2.09	1.43	1.88	1.57	2.28	1.76	2.25	1.71	1.28	2.45	2.24
	1.24	2.48	1.68	2.34	1.96	1.61	1.75	1.83	2.29	1.59	1.33	2.21
	1.64	1.48	1.86	1.74	2.37	1.91	2.49	1.78	1.78	2.24	1.88	1.12
Q	1.70	1.45	0.83	1.50	1.38	1.84	1.40	1.97	1.38	1.00	1.63	1.22
	0.88	1.71	1.48	1.82	1.50	1.38	1.36	1.40	1.97	1.44	1.00	1.65
	1.48	0.81	1.70	1.36	1.80	1.50	1.95	1.35	1.40	1.65	1.32	1.00

Initial conditions: A: $N_1 = 2, N_2(t = 500) = 0.1, N_3(t = 2500) = 0.1$; B: $N_1 = 0.8, N_2 = 0.8, N_3 = 1.2$; C: $N_1 = 0.1, N_2 = 0.1, N_3 = 0.1$; D: $N_1 = 0.1, N_2 = 0.11, N_3 = 0.12$

Figure 3:
$D = 0.5, r = 1, S_1 = S_2 = S_3 = 10$
The first column corresponds to a Droop model with species parameters:

K			**V**			**Q**		
0.96	0.28	0.96	2.21	2.22	2.35	1.40	1.88	1.20
1.07	0.72	0.32	2.44	2.00	2.49	1.18	1.40	2.04
0.60	0.84	0.86	2.71	2.25	2.11	1.93	1.21	1.40

The second column corresponds to a Monod model that results in the same community equilibrium as the Droop model ($\hat{N}_i, \hat{Q}_{ji} = c_{ji}, \hat{R}_j$):

H			**C**		
1.67	1.58	1.00	2.80	3.80	3.00
1.01	1.69	1.44	2.98	2.80	4.20
1.48	0.97	1.70	4.00	3.00	2.80

Initial conditions: are set equal to the species equilibrium values $N_1 = 1.03, N_2 = 0.99, N_3 = 1.00$ in both cases.
Figure 4:
$D = 0.25, r = 1, S_1 = 8, S_2 = 12, S_3 = 14, S_4 = 9, S_5 = 10$

K					**V**					**Q**				
1.51	1.50	1.80	0.50	1.60	1.52	1.50	2.80	1.50	1.50	0.51	0.47	0.86	0.38	0.30
1.60	1.51	1.50	1.70	0.50	1.50	1.52	1.50	2.80	1.50	0.30	0.51	0.47	0.86	0.38
0.50	1.60	1.50	1.50	1.71	1.50	1.50	1.53	1.50	2.80	0.38	0.30	0.51	0.47	0.86
1.70	0.50	1.60	1.51	1.50	2.80	1.50	1.50	1.51	1.50	0.86	0.38	0.30	0.50	0.47
1.50	1.70	0.50	1.60	1.51	1.50	2.80	1.50	1.50	1.51	0.47	0.86	0.38	0.30	0.50

Initial conditions: $N_1 = 0.1, N_2 = 0.11, N_3 = 0.12, N_4 = 0.13, N_5 = 0.14$
Figure 5:
$D = 0.25, r = 1, S_1 = 8, S_2 = 12, S_3 = 14, S_4 = 9, S_5 = 10$

K					**V**					**Q**				
1.51	1.50	1.80	0.50	1.60	1.38	1.36	2.55	1.36	1.36	0.51	0.47	0.86	0.38	0.30
1.60	1.51	1.51	1.70	0.50	1.36	1.38	1.36	2.55	1.36	0.30	0.51	0.47	0.86	0.38
0.50	1.60	1.50	1.15	1.71	1.36	1.36	1.39	1.36	2.55	0.38	0.30	0.51	0.47	0.86
1.70	0.50	1.60	1.51	1.50	2.55	1.36	1.36	1.37	1.36	0.86	0.38	0.30	0.50	0.47
1.50	1.70	0.50	1.60	1.51	1.36	2.55	1.36	1.36	1.37	0.47	0.86	0.38	0.30	0.50

Initial conditions: $N_1 = N_2 = N_3 = N_4 = 1$ but in A: $N_5 = 0.9$; B: $N_5 = 1.5$; C: $N_5 = 1.8$
Figure 6:
$D = 0.25, r = 1, S_1 = 6, S_2 = 10, S_3 = 14$

	A				B					C: insert i=5 in B(*)
K	0.11	0.02	0.08	0.15	0.22	0.26	0.24	0.07	0.05	0.06
	0.05	0.13	0.08	0.04	0.25	0.33	0.23	0.10	0.01	0.12
	0.07	0.01	0.13	0.16	0.26	0.01	0.10	0.35	0.04	0.06
V	0.21	0.30	0.20	0.57	0.26	0.49	0.27	0.48	0.09	0.14
	0.37	0.44	0.49	0.45	0.56	0.63	0.67	0.52	0.71	0.27
	0.67	0.41	0.56	0.95	0.98	0.42	0.51	1.26	0.63	0.28
Q	0.48	0.83	0.33	1.20	0.47	0.79	0.24	1.20	0.23	0.28
	0.73	0.95	1.17	1.14	0.47	0.95	1.14	1.09	2.01	0.60
	1.65	1.14	1.20	1.39	1.55	1.10	1.18	1.16	1.68	0.56

Initial conditions: A: $N_1 = 0.1, N_2 = 0.11, N_3 = 0.12, N_4(6000) = 0.1$; B: $N_1 = 0.1, N_2 = 0.11, N_3 = 0.12, N_4 = 0.13, N_5 = 0.14$; C: $N_1 = 0.1, N_2 = 0.11, N_3 = 0.12, N_4 = 0.13, N_5 = 0.14, N_6(5000) = 0.1$ (*: Take the matrices in B, copy the 5th column to the 6th, and substitute the 5th).

Supplement

File list

- `scanning.tar.gz` - Compressed archive with the C source code files
- `README` - Instructions for compilation and use

Description This program 'xscanning' performs a given number of numerical integrations for the multispecies Droop model. The simulation parameters are described in the `header.h` file, and can be modified there. These parameters include: BAND : the dominant band of the R-star matrix MAX : the number of species(=resources) RRANGE: the range from which the R-star values are chosen FRANGE: the range from which the F-hat values are chosen RUNS : the number of runs TIME : the number of days per run The program takes care of the generation of V, K and Q matrices according to the rules described in Appendix A. The source code is written in ANSI-C and makes use of the GNU Scientific Library for numerical integration of ordinary differential equations `http://www.gnu.org/software/gsl/`. More details in the README file.