

An extended patch-dynamic framework for food chains in fragmented landscapes

Jinbao Liao^{1,*}, Jiehong Chen¹, Zhixia Ying², David E. Hiebeler³ & Ivan Nijs⁴

¹Ministry of Education's Key Laboratory of Poyang Lake Wetland and Watershed Research,
Jiangxi Normal University, Ziyang Road 99, 330022 Nanchang, China

²College of Life Science, Key Laboratory of Poyang Lake Environment and Resource Utilization,
Ministry of Education, Nanchang University, Nanchang 330029, China

³Department of Mathematics and Statistics, University of Maine, 333 Neville Hall, Orono, ME
04469, USA

⁴Centre of Excellence Plant and Vegetation Ecology, University of Antwerp (Campus Drie
Eiken), Universiteitsplein 1, 2610 Wilrijk, Belgium

***Corresponding author:** Dr. Jinbao Liao (jinbaoliao@163.com)

Tel.: +86-(0)791-88133622

Fax: +86-(0)791-88120538

Supplementary A

Effects of patch availability and connectivity on average patch cluster size

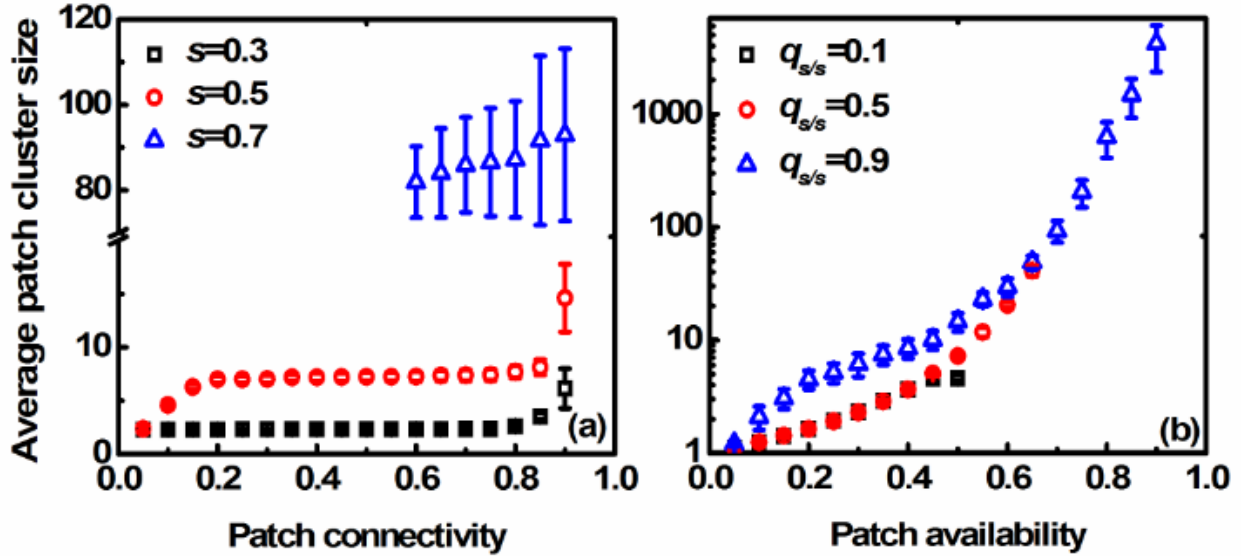


Figure S1. Effects of patch availability (s) and connectivity ($q_{s/s}$) on average patch cluster size (average $\pm$ SD of 100 replicates) in fragmented landscapes with a square lattice of size $L \times L = 100 \times 100$ cells. In each replicate, the landscape is regenerated with the same properties (i.e., keeping s and $q_{s/s}$ constant). Left panel (a): average patch cluster size by varying $q_{s/s}$ at fixed s ($=0.3, 0.5, 0.7$); and right panel (b): average patch cluster size by varying s at fixed $q_{s/s}$ ($=0.1, 0.5, 0.9$). Again, the range of patch connectivity shrinks with increasing patch availability (see equation (1)).

Supplementary B

Deriving the dynamics of $q_{l/1}$ and $q_{u/1}$

According to the definition of local density, we have $q_{l/1} = \rho_{11} / \rho_1$ and $q_{u/1} = \rho_{u1} / \rho_1$. By differentiation, we thus obtain

$$\frac{dq_{l/1}}{dt} = \frac{1}{\rho_1} \frac{d\rho_{11}}{dt} - \frac{q_{l/1}}{\rho_1} \frac{d\rho_1}{dt} \quad (B1)$$

and

$$\frac{dq_{u/1}}{dt} = \frac{1}{\rho_1} \frac{d\rho_{u1}}{dt} - \frac{q_{u/1}}{\rho_1} \frac{d\rho_1}{dt}. \quad (B2)$$

To completely construct the dynamics of $q_{l/1}$, we first derive the dynamics of ρ_{11} as

$$\frac{d\rho_{11}}{dt} = 2c_1\rho_{s1}\left(\frac{1}{z} + \frac{z-1}{z}q_{l/s1}\right) - 2\rho_{11}(e_1 + \mu_{21}\rho_{1,2} / \rho_1), \quad (B3)$$

where the first term is the state transition from the pair s-1 (or 1-s) to 1-1 via local dispersal (s-empty suitable patches, and 1-suitable patches occupied by species 1), in which the pairs s-1 and 1-s are distinguished though their pair densities are equal. This change can be achieved by one of the two pathways: the colonization of the s-patch from the 1-patch in an s-1 pair or the colonization of this s-patch from one of its (z-1) other neighbours. The latter pathway should be multiplied by $q_{l/s1}$, which is the conditional probability that another 1-patch neighbours the s-patch in a s-1 pair (using the von Neumann neighbourhood with z=4). The second term represents the loss of the pairs 1-1 $\rightarrow$ 1-s (or s-1), due to intrinsic extinction or top-down

predation. As predator species 2 with global dispersal can randomly establish only in the suitable patches with prey species 1, we can directly adopt the ratio $\rho_{1,2} / \rho_1$ to represent the proportion of prey species that may go extinct due to the top-down predation.

Similarly, we can construct the dynamics of ρ_{1u} as follows

$$\frac{d\rho_{1u}}{dt} = c_1 \rho_{su} \frac{z-1}{z} q_{1/su} - \rho_{1u} (e_1 + \mu_{21} \rho_{1,2} / \rho_1), \quad (\text{B4})$$

in which the first term on the right side denotes the transition rate of $s-u \rightarrow 1-u$ by colonizing the empty s -patch in a $s-u$ pair from one of its $z-1$ other neighbours, and the factor $q_{1/su}$ is the conditional probability that a 1-patch is the neighbour of a s -patch in the $s-u$ pair. The second term is the loss of the pairs $1-u \rightarrow s-u$ because of intrinsic extinction or top-down predation.

In view of the pair approximation principle, we obtain the following constraints

$$\begin{cases} q_{i/jk} \approx q_{i/j} \\ q_{s/1} = 1 - q_{u/1} - q_{1/1} \\ q_{1/s} = \rho_1 (1 - q_{u/1} - q_{1/1}) / (1 - u - \rho_1) \\ \rho_{su} = u q_{s/u} = u (1 - q_{u/u} - \rho_1 q_{u/1} / u) \end{cases}, \quad (\text{B5})$$

In a fragmented landscape, we have

$$\begin{cases} u + s = 1 \\ q_{u/u} = 1 - q_{s/u} = \frac{1 - 2s + s q_{s/s}}{1 - s}, \end{cases} \quad (\text{B6})$$

where patch availability s and patch connectivity $q_{s/s}$ here are constant for a given landscape, including the already occupied s -patches.

58 Combining equations (B1-B6) with (2a), we thus obtain the dynamics of $q_{l/1}$ and $q_{u/1}$ as

$$59 \quad \left\{ \begin{aligned} \frac{dq_{l/1}}{dt} &= c_1 (1 - q_{l/1} - q_{u/1}) \left(\frac{1}{2} + \frac{3}{2} \rho_1 \frac{1 - q_{u/1} - q_{l/1}}{s - \rho_1} - q_{l/1} \right) - q_{l/1} (e_1 + \mu_{21} \cdot \rho_{1,2} / \rho_1) \end{aligned} \right. \quad (B7)$$

$$\left\{ \begin{aligned} \frac{dq_{u/1}}{dt} &= c_1 \left(\frac{3}{4} \cdot \frac{s - sq_{s/s} - \rho_1 q_{u/1}}{s - \rho_1} - q_{u/1} \right) \cdot (1 - q_{u/1} - q_{l/1}) \end{aligned} \right. \quad (B8)$$

60

Supplementary C

Source code - EPDM model for a simple food chain in fragmented landscapes

```

function y=islandssystem(t,x)
global c1 c2 c3 pe1 qe1e1 e1 e2 e3 u21 u32
y=zeros(5,1);
y(1)=c1*x(1)*(1-x(4)-x(5))-e1*x(1)-u21*x(2);
y(2)=c2*x(2)*(x(1)-x(2))*qe1e1-e2*x(2)-e1*x(2)-u21*x(2)-u32*x(3);
y(3)=c3*x(3)*(x(2)-x(3))-(e1+e2+e3+u21+u32)*x(3);
y(4)=c1*(1-x(4)-x(5))*(0.5+1.5*x(1)*(1-x(4)-x(5)))/(pe1-x(1))-x(4)*(e1+u21*x(2)/x(1));
y(5)=(0.75*c1*(1-pe1)*((1-x(1)*x(5))/(1-pe1)-(1-2*pe1+pe1*qe1e1)/(1-pe1))/(pe1-x(1)))-c1*x(5))*(1-x(4)-x(5));
-----
clear
global c1 c2 c3 pe1 qe1e1 e1 e2 e3 u21 u32
c1=1; c2=1; c3=1;%species colonization rate
e1=0.05; e2=0.05; e3=0.05;%species intrinsic mortality rate
u21=0.025; u32=0.025;%top-down extinction rate
tn=[0,10000]; y0=[0.01 0.005 0.0025 0.08 0.0005]; %Initial parameter values
axispe1=0.01:0.01:0.99; axisqe1e1=0.01:0.01:0.99; % range of patch availability and connectivity
[X,Y]=meshgrid(axispe1,axisqe1e1);%pe1-patch availability and qe1e1-patch connectivity
[nn,mm]=size(X);%nn – connectivity of unsuitable patches and mm - patch loss
RDensity=zeros(nn,mm)-1;%resource
CDensity=zeros(nn,mm)-1;%consumer
PDensity=zeros(nn,mm)-1;%predator
RRDensity=zeros(nn,mm)-1;%local density of resource species
REDensity=zeros(nn,mm)-1;%conditional probability  $q_{u/1}$ 
Periodicarea=zeros(nn,mm)-1;% test system oscillation or whether system reaches stability
for n1=1:nn
    for n2=1:mm
        pe1=axispe1(n2);qe1e1=axisqe1e1(n1);
        if (pe1<0.5)||((pe1>=0.5&&qe1e1>2-1/pe1) % constraint for patch availability and connectivity
[t,y]=ode45('islandssystem',tn,y0); %algorithm of EPDM equations
if max(y(end-100:end,1))>mean(y(end-100:end,1))+0.0005
            Periodicarea(n1,n2)=1;
        else
            Periodicarea(n1,n2)=0;
    end
    RDensity(n1,n2)=mean(y(end-100:end,1));
    RRDensity(n1,n2)=mean(y(end-100:end,4));
    REDensity(n1,n2)=mean(y(end-100:end,5));
    CDensity(n1,n2)=mean(y(end-100:end,2));
    PDensity(n1,n2)=mean(y(end-100:end,3));
    else
        RDensity(n1,n2)=NaN;
        RRDensity(n1,n2)=NaN;
    end
end

```

Supplementary Information

```
109 REDensity(n1,n2)=NaN;
110 CDensity(n1,n2)=NaN;
111 PDensity(n1,n2)=NaN;
112     end
113 end
114 end
115
116 RDensityi=find(RDensity<0.0001);%extinction threshold of resource
117 RDensity(RDensityi)=0;
118 CDensityi=find(CDensity<0.0001);%extinction threshold of consumer
119 CDensity(CDensityi)=0;
120 PDensityi=find(PDensity<0.0001);%extinction threshold of predator
121 PDensity(PDensityi)=0;
122
123 figure,surf(X,Y,Periodicarea),view(0,90),colorbar
124 xlabel('pe1'),ylabel('qe1e1')
125 figure,surf(X,Y,REDensity),view(0,90),colorbar
126 xlabel('pe1'),ylabel('qe1e1')
127 figure,surf(X,Y,CDensity),view(0,90),colorbar
128 xlabel('pe1'),ylabel('qe1e1')
129 figure,surf(X,Y,PDensity),view(0,90),colorbar
130 xlabel('pe1'),ylabel('qe1e1')
131
132 Z=REDensity;
133 ii0=find(RDensity<0.0001);%resource does not exist
134 ii1=find((REDensity>0.0001)&(CDensity<0.0001));%resource exists but consumer goes extinct
135 ii2=find((REDensity>0.0001)&(CDensity>0.0001)&(PDensity<0.0001));%resource & consumer coexist
136 but predator goes extinct
137 ii3=find((REDensity>0.0001)&(CDensity>0.0001)&(PDensity>0.0001));%All species coexistence
138 Z(ii0)=0; Z(ii1)=1; Z(ii2)=2; Z(ii3)=3;
139 figure,surf(X,Y,Z),view(0,90),colorbar
140 xlabel('pe1'),ylabel('qe1e1')
141
```

142

Supplementary D

143 *Source code - CA simulation for a simple food chain in fragmented landscapes*

```

144 function [Predator,Consume,resource,pp,pc,pr]=stept(A,Predator,Consume,resource)
145 c1=1; c2=1; c3=1; %species colonization rate
146 e1=0.025; e2=0.025; e3=0.025; %species intrinsic mortality rate
147 u21=0.05; u32=0.05; %Top-down extinction rate
148
149 %patch cluster reorganization
150 [n1,n2]=size(A);
151 output = bwconncomp(A,4); %based on 4-nearest neighbourhood
152 CC=output.PixelIdxList; % patch locations in the landscape
153 Npatches=output.NumObjects;%patch numbers
154
155 %caculate the global density of Predator, Consumer, Resource
156 pp=0;pc=0;pr=0;pe2=0;
157 for ii=1:n1
158     for jj=1:n2
159         if Predator(ii,jj)==2
160             pp=pp+1;
161         end
162         if Consume(ii,jj)==2
163             pc=pc+1;
164         end
165         if resource(ii,jj)==2
166             pr=pr+1;
167         end
168         if A(ii,jj)==0
169             pe2=pe2+1;
170         end
171     end
172 end
173 pp=pp/(n1*n2);pc=pc/(n1*n2);
174 pr=pr/(n1*n2); pe2=pe2/(n1*n2);
175
176 Consume2=Consume;
177 resource2=resource;
178 Predator2=Predator;
179
180 % %%%%Top Predator
181 % state transition from 2 (predator) to 1(unoccupied by predator)
182 % and transition from 1(unoccupied by predator) to 2(predator)
183 for ii=1:n1
184     for jj=1:n2
185         if Predator2(ii,jj)==2
186             p1=rand(1); p2=rand(1); p3=rand(1); p4=rand(1); p5=rand(1);
187             if (p1<e1)|(p2<e2)|(p3<e3)|(p4<u21)|(p5<u32) %intrinsic mortality and top-down extinction
188                 Predator(ii,jj)=1;
189             end

```


Supplementary Information

```

190         end
191         if (Predator2(ii,jj)==1)&(Consume2(ii,jj)==2)&(resource2(ii,jj)==2)
192             p1=rand(1);
193             if p1<1-(1-0.0001)^(c3*pp*10000) %patch occupancy
194                 Predator(ii,jj)=2;
195             end
196         end
197     end
198 end
199
200 %%%%%%%%%Intermediate consumer
201 % state transition from 2 (consumer) to 1(unoccupied by consumer)
202 for ii=1:n1
203     for jj=1:n2
204         if Consume2(ii,jj)==2
205             p1=rand(1); p2=rand(1); p3=rand(1); p4=rand(1);
206             if (p1<e2)|(p2<e1)|(p3<u21)|(Predator2(ii,jj)==2 & p4<u32) %
207                 Consume(ii,jj)=1;
208             end
209         end
210     end
211 end
212
213 % transition from 1(unoccupied by consumer) to 2(consumer)
214 for ii= 1:Npatches % Npatches is the patch cluster number
215     xylabel=CC{ii}; % the ii-th patch cluster location in the 2-D matrix or the 1-D matrix
216     [nvalue,nvalue2]=size(xylabel); % nvalue is the patch cluster size of patch cluster i
217     region_C=sum(Consume2(xylabel)-1);% region_C is the consumer number in the ii-th patch cluster
218     region_R=sum(resource2(xylabel)-1);% region_R is the resource number in the ii-th patch cluster
219     region_Rdensity=region_R/nvalue;% region_Rdensity is the local density in the ii-th patch cluster
220     region_Cdensity=region_C/nvalue;
221     for jj=1:nvalue
222         jj1=xylabel(jj);
223         if Consume2(jj1)==1 & resource2(jj1)==2
224             p1=rand(1);
225             if p1<1-(1-1/nvalue)^(c2*region_C)
226                 Consume(jj1)=2;
227             end
228         end
229     end
230 end
231
232 %%%%%%%%% %Resource
233 % transition from 1(unoccupied by resource) to 2(resource)
234 for ii=1:n1
235     for jj=1:n2
236         if ii==1&&jj==1
237             x=[resource2(100,jj),resource2(ii+1,jj),resource2(ii,100),resource2(ii,jj+1)];
238             end
239         if ii==1&&jj==100
240             x=[resource2(100,jj),resource2(ii+1,jj),resource2(ii,jj-1),resource2(ii,1)];

```

Supplementary Information

```
241     end
242     if ii==100&&jj==100
243     x=[resource2(ii-1,jj),resource2(1,jj),resource2(ii,jj-1),resource2(ii,1)];
244     end
245     if ii==100&&jj==1
246     x=[resource2(ii-1,jj),resource2(1,jj),resource2(ii,100),resource2(ii,jj+1)];
247     end
248     if ii==1&&(jj>1&&jj<100)
249     x=[resource2(100,jj),resource2(ii+1,jj),resource2(ii,jj-1),resource2(ii,jj+1)];
250     end
251     if ii==100&&(jj>1&&jj<100)
252     x=[resource2(ii-1,jj),resource2(1,jj),resource2(ii,jj-1),resource2(ii,jj+1)];
253     end
254     if jj==100&&(ii>1&&ii<100)
255     x=[resource2(ii-1,jj),resource2(ii+1,jj),resource2(ii,jj-1),resource2(ii,1)];
256     end
257     if jj==1&&(ii>1&&ii<100)
258     x=[resource2(ii-1,jj),resource2(ii+1,jj),resource2(ii,100),resource2(ii,jj+1)];
259     end
260     if ii>1&&ii<100&&jj>1&&jj<100
261     x=[resource2(ii-1,jj),resource2(ii+1,jj),resource2(ii,jj-1),resource2(ii,jj+1)];
262     end
263     x1=0;
264     for kk=1:4
265         if x(kk)==2
266             x1=x1+1;% the number of occupied sites
267         end
268     end
269     if resource2(ii,jj)==1
270     p1=rand(1);
271     if p1< 1-(1-1/4)^(x1*c1)
272     resource(ii,jj)=2;
273     end
274     end
275 end
276 end
277
278 % transition from 2(resource) to 1(unoccupied by resource)
279 for ii=1:n1
280     for jj=1:n2
281         if resource2(ii,jj)==2
282             p1=rand(1);p2=rand(1);
283             if (p1<e1)|(Consume2(ii,jj)==2&p2<u21)
284                 resource(ii,jj)=1;
285             end
286         end
287     end
288 end
289
290 for ii=1:n1
291     for jj=1:n2
```

Supplementary Information

```
292 if resource(ii,jj)==1
293 Consume(ii,jj)=1;
294 Predator(ii,jj)=1;
295 end
296 if Consume(ii,jj)==1
297 Predator(ii,jj)=1;
298 end
299 end
300 end
301
302 a=load('data(0.5,0.5).txt');% Fragmented landscape by taking s=0.5 and qs/s=0.5 for example
303 %see landscape generation algorithm in Hiebeler 2000 (Ecology) and 2007 (J. Math. Biol.)
304 n1=100;n2=100;%lattice size
305 A= reshape(a,n1,n2);%
306 % The initial value of resource, consume, predator
307 resource=A;%resource value 0,1,2 respectively expresses non-habitat, suitable habitat but without
308 resource, and suitable site with resource
309 Consume=A;% Consume value 0,1,2 respectively denotes nonhabitat, suitable habitat but without
310 consumer, and suitable site with consumer
311 Predator=A;% Predator value 0,1,2 respectively denotes nonhabitat, suitable habitat but without predator,
312 and suitable site with predator
313 P_initial_resource=0.5;%initial resource occupancy
314 P_initial_Consume=0.5;%initial consumer occupancy
315 P_initial_Predator=0.5;%initial predator occupancy
316     for ii=1:n1
317         for jj=1:n2
318             if A(ii,jj)==1
319                 p1=rand(1); p2=rand(1); p3=rand(1);
320                 if P_initial_resource>p1
321                     resource(ii,jj)=2;
322                 end
323                 if (P_initial_Consume>p2)&(resource(ii,jj)==2)
324                     Consume(ii,jj)=2;
325                 end
326                 if (P_initial_Predator>p3)&(Consume(ii,jj)==2)
327                     Predator(ii,jj)=2;
328                 end
329             end
330         end
331     end
332
333 nn=100000;%time steps
334 Pdensity=zeros(1,nn);Cdensity=zeros(1,nn);rdensity=zeros(1,nn);
335 % predator, consumer, resource over time
336
337 for ii=1:nn
338 [Predator,Consume,resource,pp,pc,pr]=stept(A,Predator,Consume,resource);
339 Pdensity(ii)= pp; Cdensity(ii)=pc;rdensity(ii)=pr;
340 end
341
342 %average over the final 10000 time steps
```

Supplementary Information

```
343 mean(rdensity(90000:100000))
344 mean(Cdensity(90000:100000))
345 mean(Pdensity(90000:100000))
```